Scalability and Satisfiability of Quality-of-Information in Wireless Networks

Scott T. Rager* Ertugrul N. Ciftcioglu† Ram Ramanathan‡ Thomas F. La Porta* Ramesh Govindan§
*The Pennsylvania State University, University Park, PA 16802
†IBM Research, Yorktown Heights, NY 10598
‡Raytheon BBN Technologies, Cambridge, MA 02138
§University of Southern California, Los Angeles, CA 90089
Email: rager@psu.edu, enciftci@us.ibm.com , ramanath@bbn.com, tlp@cse.psu.edu, ramesh@usc.edu

***Abstract*—Quality of Information (QoI) provides a context-dependent measure of the utility that a network delivers to its users by incorporating non-traditional information attributes. Quickly and easily predicting performance and limitations of a network using QoI metrics is a valuable tool for network design. Even more useful is an understanding of how network components like topology, bandwidth, protocols, etc. impact these limitations. In this paper, we develop a QoI-based framework that can provide this understanding of limitations and impact by modeling the various contributors to delay in the network, including channel rate and contention, competing traffic flows, and multi-hop propagation effects, and relating them to QoI requirements, especially completeness and timeliness. Analysis shows that large tradeoffs exist between network parameters, such as QoI requirements, topology, and network size. Simulation results also provide evidence that the developed framework can estimate network limits with high accuracy. Finally, this work also introduces *scalably feasible QoI regions*, which provide upper bounds on QoI requirements that can be supported for certain network applications.**

I. Introduction

Traditionally, approaches to studying network scalability and performance limitations have been focused on finding theoretical limits on throughput and delay. Additionally, much of this work focuses on one network protocol or parameter at a time. While this research is useful for determining upper bounds on performance for specific cases, it lacks several considerations that we address in this work.

First, only considering delay or network throughput does not capture the impact of these metrics on the data's usefulness. In the same manner, simply using throughput as a metric implies that all delivered data is equally useful. As we show in Section III, this assumption is unrealistic for many applications. For these reasons, we adopt *Quality of Information* (QoI), which can include a number of information attributes (many of which are context-dependent), such as completeness, diversity, credibility, creation time, and timeliness, as our measure of network performance. Specifically, in this work, we focus on satisfiability of completeness and timeliness, explained in detail in Section III.

The second contribution differentiating this work from previous analysis is that we focus on providing a framework that can be adjusted to determine scalability and QoI satisfiability for any instance of a network. The wide applicability and easy reuse of this framework make it easy to compare protocols, topologies, traffic models, etc., without creating extensive simulation or experimentation testbeds, which is a much more difficult task.

Finally, in contrast to discovering theoretical, asymptotic limits, this framework seeks to quickly obtain an accurate estimate of a network's abilities. While the goal of this paper is not in optimizing network performance, we show in Section VI that this model can also be used to quickly and easily understand the impact of parameters and design choices. This ability provides a secondary benefit to network designers of allowing them to compare networks and identify tradeoffs.

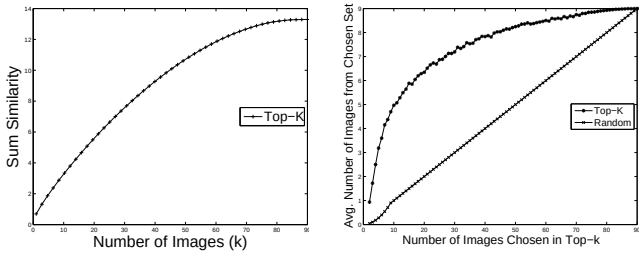
As an example, imagine given the task of deploying a wireless sensor network for a particular application. Given a proposed network with a defined signature describing its size, parameters, protocols, and expected traffic, what is the level of QoI requirements it can support? Alternatively, consider the converse: Given a certain QoI that is desired by users of a network, what is the maximum number of nodes that the network can support? Which has a bigger impact on this scalability, the imposed data requirements, or the strict timeliness requirements?

Currently, no framework exists that provides a methodology to predict scalability and performance with respect to QoI requirements. In this paper, our contribution is to provide such a framework.

II. Related Work

The scalability model derived in this work is inspired by the symptotic scalability framework outlined in [1], which has been previously applied to content-agnostic static networks [2] and mobile networks [3]. Other works characterize the capacity of wireless networks, like [4], [5], but all do so differently by considering how networks scale asymptotically or by analyzing specific network instances instead of developing a general model. Experimental techniques, like Response Surface Methodology [6], for example, may be applied to solve the problem we do, but these require complex test beds instead of a compact mathematical framework.

Research was sponsored by the U.S. Army Research Laboratory under the Network Science Collaborative Technology Alliance, Agreement Number W911NF-09-2-0053.



(a) Top-K: Sum Similarity (b) Top-K: Avg. Match Target

Fig. 1. Completeness metrics for the three image selection algorithms. Each exhibits a diminishing return as more images are added.

A large number of works provide definitions for Quality of Information and frameworks utilizing it. We will address only the most relevant ones here. Primarily, QoI has been used in scheduling and has been considered from a number of various angles routing [7], scheduling/rate control [8], [9], and impact on usage of network resources [10]. Our focus is on a broader scale here, though, modeling scalability and limitations of an entire network.

Additionally, [11] outlines a framework called Operational Information Content Capacity, which describes the obtainable region of QoI, a notion similar to the *scalably feasible QoI region* developed here. These approaches use a general network model, though, and do not provide any method for determining the possible size of the network or impact of various network design choices like medium access protocols.

In Section III, we use similarity-based image collection as an example of an application that is best evaluated using QoI. This application has previously been considered in [12] and [13]. Our scope is greater than that of [12], which does not consider attributes of timeliness, nor the consideration of transmission rates and network topology. We use the same similiary-based image selection algorithms as in [13], but provide new methods of quantifying QoI from them.

III. QoI Model

QoI is a metric that can be defined for an application to give a more meaningful value to information by capturing attributes such as timeliness, age/freshness, completeness, accuracy, precision, etc. For example, information that contributes to a decision-making process may only be useful if it arrives before the decision must be made, or it may have varying usefulness based on how similar or dissimilar it is to other data already collected.

The specific details of which attributes are considered and how they contribute to QoI is application-dependent. The chosen QoI metrics are stored as a vector associated with a data item. Here, as in [14], we specify a vector of minimum values for each QoI metric, and data is evaluated based on whether it satisfies all of the QoI requirements or not. We use this approach to establish the edges of QoI satisfiability for the vector of metrics, which defines the boundaries of maximum achievable QoI regions in the metric space.

We choose to use two QoI attributes, one that is time-based and one that is information-content-based. The first attribute is timeliness, T, of data. For the second attribute, we present a notion of *completeness*, C, which we show can be defined multiple ways, depending on the context. Together, a QoI requirement of $\mathbf{q} = \{C, T\}$ specifies a quantity of data that must be delivered as well as a deadline by which it must arrive to be useful. Since completeness is a new concept, we explain three image selection algorithms and show how they can be evaluated with completeness.

A. Image Selection Algorithms

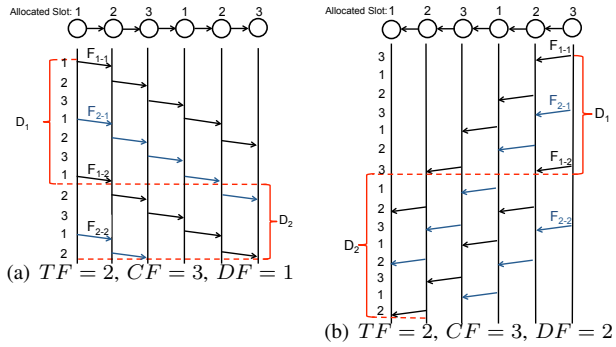
As a motivating example, we choose a network in which nodes store photographs that are to be exchanged or collected at one or more data sinks. This example covers surveillance missions of military tactical networks or camera sensor networks. Without adopting any one specific application, we present three query algorithms from [13], one that helps return images similar to a target, and two that help return a diverse set of images. We also present ways to measure the completeness of each, providing a quantified value of a QoI metric. All similarity values are calculated by taking the *Tanimoto Similarity* [16] of vectors describing the photo characteristics, found using a technique called Color and Edge Directivity Descriptor (CEDD) [15].

1) Selecting Similar Images: The first type of query we introduce occurs when a user already has one image of a particular area or object of interest and would like to obtain similar images to get a more complete view of that specific scene or object. For example, if a user has a picture of an unknown suspicious person entering a building, but the person is not identifiable from that image, it would be useful to collect more images that are similar to that one with the possibility that another picture may have a better view of the person in question that can be used for identification or more context. Called **Top-K**, the query algorithm used for this application will choose the k images with the most similarity with respect to the target image.

We can evaluate the completeness of the result in one of two ways. First, we can use the similarity of the images as a value representing each image's effectiveness in providing a more complete view of the target scene. If we sum the similarity of all k images returned by the algorithm, we get a representation of completeness, which we naturally call *Sum Similarity*. Second, if the environment is able to be partitioned into n distinct area or settings, then completeness can be defined as the fraction of images that are in the same set as the target image.

2) Experimental Results: To provide example values of these completeness metric definitions, experiments applying each query algorithm were run on a set of pictures taken at $n = 9$ different settings around the Penn State campus. The three algorithms were run $1,000$ times each on a random sampling of 90 images in each trial. Average results for each completeness metric are shown in Figures 1(a) to **??**.

All of these figures exhibit the diminishing returns of completeness as more images are collected. This effect visually shows how QoI differs from throughput. As seen in these graphs, transmission of successive images does not



(a) $TF = 2$, $CF = 3$, $DF = 1$ (b) $TF = 2$, $CF = 3$, $DF = 2$

Fig. 2. Example of line network using TDMA highlighting source of delays. (Node labels are TDMA slot assignments.)

result in a linear gain in completeness. For example, in Figure 1(b), it is evident that a value of only $k \approx 10$ is needed to collect 5 images matching the target content, while collecting an additional 2 from the same set usually requires collecting over twice that number of pictures. Similarly, Figure **??** shows that jumping from $k = 10$ to $k = 20$, the likelihood of capturing at least one image of every setting grows substantially from just over 10% to approximately 90%. To approach probabilities close to gaining that final 10%, however, requires a jump to $k \approx 30$.

IV. QoI Scalability

Given the nonlinear returns of completeness and importance of timeliness outlined in the previous section, we contend that simply establishing the highest average supportable rate should not be the only goal of a QoI-aware network. With this knowledge, we set the goal of determining the capacity of a network (and relatedly, the scalability achievable) with respect to *QoI requirements*, instead of the maximum throughput.

A. QoI Satisfiability Framework

In order to establish the framework, we examine an arbitrary flow, F_1, in the network that has a QoI requirement of $\mathbf{q} = \{C, T\}$, where C is the minimum required completeness metric of choice, such as sum similarity as explained above, and T is the required timeliness. This flow will have a data size requirement, which is given by a chosen QoI function $Q(C)$ like those discussed in Section III. Using the example applications from III, for example, $Q(C)$ can return the number of images, k_{req}, required to achieve the requested completeness C according to established relations like those in Section III. Assuming each image has an average size of I_S, then we can also use B to describe the total number of bits required by the flow, $B = k_{req} * I_S$. To match realistic network implications, we assume this data will be transmitted in a series of packets with size P_S bits each. The number of packets per flow, then, is simply $P_N = \lceil B/P_S \rceil$. We assume that each node in the network can transmit at W bits per second when it is allocated media access.

Our goal is to establish the limits at which this arbitrary flow on average can no longer be completed with the QoI requirements satisfied. We build and explain our model for achieving this goal by working through an example TDMA line network, a portion of which is shown in Figures 2(a) and 2(b). (We discuss addressing non-TDMA networks in Section **??**.) In this network, we assume a simple 3-slot TDMA scheme, which allows each node equal time access to the medium and removes any potential interference or hidden terminal issues. The TDMA Slot assignments are labeled above each node. For simplicity, we will assume that one slot is appropriately sized to transmit a single packet, i.e., $T_{slot} = P_S/W$ and that packets use static routes calculated beforehand such that the overhead is not a consideration here.

Now, two factors, D_1 and D_2, contribute to the total delay of completing F_1. The first contributor, D_1, is the end to end delay incurred by sending the B bits across the entire path. To quantify this delay, we must consider several factors beyond just the available bandwidth and number of packets. First, each node can only utilize its allotted channel time, creating a Channel Factor of $CF = N_{frame}/N_i$, where N_i is the number of slots allocated to node i and N_{frame} is the total number of slots in each frame. In this case, $N_i = 1, \forall i$ and $N_{frame} = 3$. The second factor to consider for D_1 is the fraction of allocated slots that are utilized by the node to serve flows other than F_1 that are either originating in or being routed through nodes on the path of flow F_1. We call the total number of flows competing at node i the Traffic Factor, TF_i, of that node. For any flow, the maximum contributor to delay is the node along the path with the maximum TF_i, which we will just call TF here. Incorporating these considerations into a calculation for D_1, we achieve the following expression:

$$D_1 = T_{slot} \cdot P_N \cdot CF \cdot TF \tag{1}$$

Figure 2(a) depicts the delay of D_1 in a simple case of only two flows, F_1 and F_2, being present, in which case $TF = 2$. In this example we use flows that consist of only 2 (F_{i-j} is packet j in flow i) packets to portray the delay. In most real applications, P_N will be much larger, making D_1 a good approximation of this delay component.

The second delay that exists is due to multi-hop propagation of packets. This delay is simply the time for a single packet to traverse the path length. Note that this delay is not necessarily just the path length multiplied by T_{slot}, because of possible queuing delays and/or ordering constraints. We show here how ordering constraints impact this TDMA network. A node cannot forward a packet from the flow until it receives that packet from the previous hop. In our line network example, when the direction of the flow matches the nodes' schedule of slots $1-2-3-1-2-3$, as in Figure 2(a), each successive node receives a packet on the time slot before it is scheduled, resulting in no extra delay. For a flow in the opposite direction, though, where nodes are scheduled $3-2-1-3-2-1$, as in Figure 2(b), the first slot 1 is not utilized, because the first node scheduled in time slot 1 has not yet received a packet in the flow. Every other slot is wasted, on average, for the initialization of the flow, resulting in approximately twice the delay. We will use a term that we call the *Delay Factor*, or DF, to account for this effect where it exists.

The multi-hop propagation delay, then, is modeled by

$$D_2 = T_{slot} \cdot DF \cdot (PL - 1) \quad (2)$$

where PL is the average path length.

We note several points about this delay factor. First, in a loaded network, the nodes can and will serve other flows while awaiting the arrival of packets in this flow of focus. That utilized bandwidth does not, however, preclude this DF impact on delay for the flow of interest, F_1. Any node cannot serve F_1 until a packet from that flow has been received. Second, this delay is only accounted for once per flow because all other packets are pipelined. All other packets' delay is captured by the end to end delay, D_1. This effect is best illustrated by examining the difference between D_2 in Figures 2(a) and 2(b). Here, we see the multi-hop propagation requires twice the number of slots because every other slot is unused in F_1's propagation. To calculate a DF for an entire network, we can calculate a DF for each possible sample path and find the average of these values.

By adding the two components of delay, we can give a relation for a network that will successfully achieve an average flow's data and timeliness requirements:

$$T_{slot} \cdot P_N \cdot CF \cdot TF + T_{slot} \cdot DF \cdot (PL - 1) \leq T$$

Recalling that the time of a slot is determined by $T_{slot} = P_S/W$ and that the total number of bits required for a flow $P_S \cdot P_N = k_{req} \cdot I_S$ (where k_{req} is given by a function of required QoI), we can get the following expression:

$$W \cdot T - k_{req} \cdot I_S \cdot CF \cdot TF - P_S \cdot DF \cdot (PL - 1) \geq 0 \quad (3)$$

Every network will have its own set of parameters that can be substituted into this general formula, providing a tool to approximate limitations and sensitivity to changes in specific parameters. To show this usefulness concretely, we provide derived values for a TDMA-based wireless network with several basic topologies.

B. Example of Applying Framework

We illustrate the application of network signatures to the relationship in (3) using an N-node TDMA network with three different topologies: clique, line, and grid, also known as a "Manhattan grid." (Discussion of other network control protocols and topologies are addressed in Section **??**.) We adopt a traffic model that uses Top-K queries as an example application. We assume that all nodes have a set of collected images that are used to respond to Top-K queries. Each node produces a query with a target image and target QoI, $\mathbf{q} = \{C, T\}$, describing the required completeness (here, we use sum similarity) and timeliness, and sends it to another node chosen at random. The queried node will respond with the number of images, k_{req}, required to achieve the target sum similarity. Values for k_{req} are taken from the empirical relation in Figure 1(a).

Since our goal is to determine the point at which an average flow is no longer sustainable, we derive and use average values for TF, CF, DF, and PL for the network. In the case of TF, we use the value for the node with the largest expected TF_i since flows that are routed through this node are expected to experience that largest delay and are likely to be the first that fail to meet their timeliness requirements. Values for this example are shown in Table I, and a derivation of TF for a grid network is included in Appendix A. Details about deriving the other values are explained in detail in [1]. The following equations can be used to determine QoI and network size limitations, which will be exemplified in the next two sections:

	CF	TF	DF	PL
Clique	N-1	1	1	1
Line	3	$\frac{(N-1)^2}{2(N-2)}$	1.5	$\frac{N}{4}$
Grid	5	$\sqrt{N}$	2.5	$\frac{2}{3}\sqrt{N}$

TABLE I
CF, TF, DF, AND PL VALUES FOR EXAMPLE TOPOLOGIES

	Equation
Clique	$W \cdot T - I_S \cdot k_{req} \cdot (N-1) \geq 0$
Line	$W \cdot T - 3 \cdot I_S \cdot k_{req} \cdot \frac{(N-1)^2}{N-2} - 1.5 \cdot P_S \cdot (\frac{N}{4} - 1) \geq 0$
Grid	$W \cdot T - 5 \cdot I_S \cdot k_{req} \cdot \sqrt{N} - 2.5 \cdot P_S \cdot (\frac{2}{3}\sqrt{N} - 1) \geq 0$

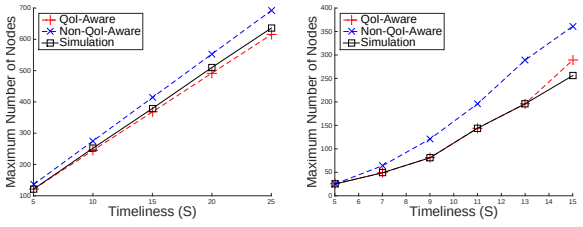
TABLE II
SCALABILITY EQUATIONS

V. VALIDATION

To show how effective estimates using this framework can be, we simulated the network topologies and traffic described above in Section IV-B in the ns3 network simulator, comparing empirical results to those generated analytically. To show the role of considering QoI, we also include *non-QoI-Aware* scalability estimates that arise from using the scalability framework in [2]. We only show a subset of results in Figure 3 to provide evidence of the effectiveness of the methodology. All results generated, however, exhibit very similar trends of proximity between empirical and the analytical values.

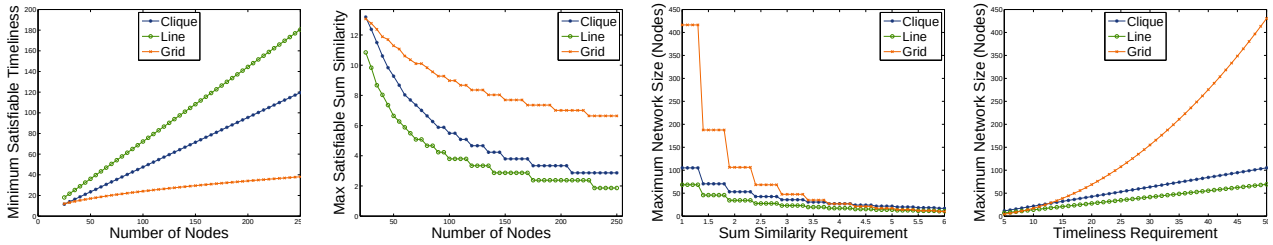
VI. IMPACT ON NETWORK DESIGN

Now that we have established a model for QoI satisfiability and scalability and shown its accuracy in predicting estimated limits, we show how it can also provide quick, but useful intuition about the impact of various network parameters. Once an equation for a network's limitations is formulated as was done for equations in Table II , it can be solved for



(a) Line Network, $I_S = 3MB$ (b) Grid Network, $I_S = 6MB$

Fig. 3. Empirical results match analytical results closely for all performed tests. Comparison to non-QoI-Aware models also show value of framework.



(a) Max Timeliness vs. Network Size (Sum Sim. = 5.0)

(b) Max Sum Sim. vs. Network Size (Timeliness = 50)

(c) Max Network Size vs. Sum Sim. (Timeliness = 10)

(d) Max Network Size vs. Timeliness (Sum Sim. = 5.0)

Fig. 4. Varying design parameters provide immediate limitations as well as evident trends and comparisons.

a single parameter to gain an understanding of its reliance on the other factors. This analysis can be done very quickly and easily by approximating when appropriate, or by using symbolic and/or numerical solvers. To exemplify this application in network design, we have applied this process to the equations in Table II and created several figures that provide insight into network limitations and tradeoffs.

Figure 4(a) shows minimum satisfiable timeliness values for different topologies of increasing size. While it may be intuitive that a grid network should be able to serve flows with lower timeliness constraints than a line network since it will have shorter average path lengths, what is not intuitive is *how much* lower of a timeliness constraint is satisfiable. Here, we see that a grid network can scale up to over 250 nodes while supporting flows with $T = 40$. A line network, on the other hand, can only scale to only approximately 20% of that network size for the same timeliness value.

Figure 4(b) visualizes sum similarity limits for network instances with timeliness fixed. Here, again, limits and trends are quickly evident. In addition, the nonlinearity of completeness requirements are manifested here as the data requirements for the highest completeness requirements are only sustainable for small networks, but as the network size increases, the impact on achievable completeness is not as dramatic.

Finally, given a topology and application with predetermined requirements of $\mathbf{q}$, we may be simply interested in how large our network can grow before its capacity to deliver the desired QoI is no longer possible. Values that answer this question are in Figures 4(c) and 4(d). Once again, the graphs show the major impact completeness and timeliness requirements can have on network size.

VII. Scalably Feasible QoI Regions

Now let us consider a special case where nodes collect a total of k_{req} images, but each image must come from a different node, perhaps because each node only has one image or maybe by design to provide increased credibility through corroboration, another contextual metric of interest in QoI-aware networks. This scenario allows us to illustrate an interesting observation.

Consider a set of QoI requirements that include completeness and timeliness, $\mathbf{q} = \{C, T\}$. As previously noted, a certain number of images are required to achieve this desired level of completeness, $Q(C) = k_{req}$. Unlike the model used in Section VI, however, each node can contribute only one image to k_{req}, which implies a minimum network size of $N \geq k_{req}$ in order to achieve the completeness outlined in the QoI requirements. On the other hand, applying the framework of Section IV to this network, we can determine the maximum network size, which we will call N_{max} here for distinction.

These two facts are important, because when $N_{max} < k_{req}$, then it is not possible to provide the QoI level $\mathbf{q}$. Hence, we say that this set of QoI requirements is infeasible, or *scalably infeasible*. This phenomenon defines the concept of a *Scalably Feasible QoI Region*, which refers to the region in which all sets of QoI pairs can be supported with the given network signature.

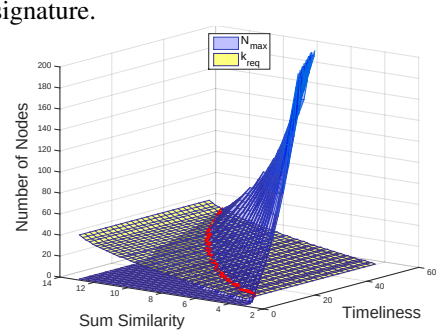


Fig. 5. The blue plane represents maximum scalability, and the yellow plane represents minimum required images. Therefore, all Sum Similarity, Timeliness pairs to the right of the red line are within the scalably feasible QoI region.

Figure 5 provides a visual representation of this region for a grid network that institutes the given traffic model assuming traffic is routed over randomly chosen shortest-path routes. Here, N_{max}, calculated for $\mathbf{q}$ pairs from $\{2.5, 1\}$ to $\{13.0, 50\}$, is shown in the graph by the blue surface. On the same graph is the number of images required, k_{req}, for each Sum Similarity requirement, shown with the yellow surface on the graph. The intersection of these two surfaces, displayed with a red line, provides the edge of the scalably feasible QoI region. In this example, all sum QoI pairs to the right of this line, i.e. the region where $N_{max} > k_{req}$, are scalably feasible.

In general, regardless of how many images a node has, it is possible to analyze the trade-off between different QoI attributes for a fixed value of maximum network size, N_{max}. Specifically, by fixing N in Table II, one can obtain T and k_{req} (and, hence, C) resulting in the set of supportable $\{C, T\}$ pairs defining a feasible region for QoI. This region

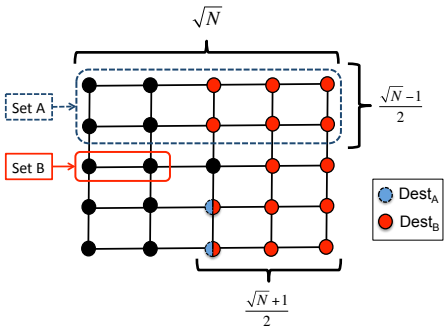


Fig. 6. Sources and destinations used in proving TF for grid networks

can be visualized by intersecting the blue scalability curve with a flat surface fixed at N_{max} instead of the yellow surface in Figure 5. The scenario given in this section goes one step further and also takes into account the network size actually required to generate/support a given QoI requirement (using the non-flat yellow surface derived from experimental results in Figure 5).

VIII. Conclusion

This work provides several contributions to the field of QoI-aware wireless networks. First, we motivated the use of completeness and timeliness as QoI attributes, providing example applications and several different ways to measure completeness in these applications. Next, we developed a framework that can be used to predict QoI and network size limits for a specific network and then validated the framework's accuracy by comparing analytical results with simulations performed in the ns3 network simulator. Examples of the impact of different network parameters were shown, providing concrete examples of the framework's usefulness in real-world applications. In addition, the concept of scalably feasible QoI regions was introduced. In addition to future work already mentioned in Section **??**, we plan to expand this framework to include consideration of more complex network control actions, such as caching and/or data compression or fusion, which are all of interest in QoI-aware networking.

Appendix A
Proof of Traffic Factor for Grid Network

We outline a simple proof for determining the traffic factor of the center node in a grid of N nodes using "Row-First, Column-Second" routing. We give the proof for when $\sqrt{N}$ is odd, but the logic follows similarly for even values. Assume each node is the source of exactly one flow and that the destination is uniformly chosen from all other $N-1$ nodes. Node i, then, has a $\frac{1}{N-2}$ chance of choosing each non-center node.

First, we consider the nodes circled in set A in Figure 6. The destinations that are relayed through the center are marked in blue in the figure. The probability of a node in set A choosing one of these destinations is $P_A = \frac{\frac{\sqrt{N}-1}{2}}{N-2}$. The total number of nodes falling into set A, including the mirror of those circled in the figure, is $N_A = \sqrt{N}\cdot(\sqrt{N}-1)$. Then, the contribution to the TF by nodes in set A is simply the product of P_A and N_A: $E[TF_A] = \frac{\frac{\sqrt{N}-1}{2}}{N-2}\cdot\sqrt{N}\cdot(\sqrt{N}-1)$

Similarly, for set B: $P_B = \frac{\frac{\sqrt{N}+1}{2}\cdot\sqrt{N}-1}{N-2}$ and $N_B = \sqrt{N}$, so the expected contribution to TF from set B is $E[TF_B] = \frac{\frac{\sqrt{N}+1}{2}\cdot\sqrt{N}-1}{N-2}\cdot 2\cdot(\frac{\sqrt{N}-1}{2})$.

Since sets A and B account for all non-center nodes in the network, the overall expected traffic factor is just the sum of $E[TF_A]$ and $E[TF_B]$, which simplifies to

$$E[TF] = \frac{\sqrt{N}(N-2)+1}{N-2} \tag{4}$$

which is effectively $\sqrt{N}$ for large N.

References

[1] R. Ramanathan, Ciftcioglu E., A. Samanta, Urgaonkar R., and T.L. Porta. An approximate model for analyzing real-world wireless network scalability. Technical report, Raytheon BBN Technologies, 2015. http://www.ir.bbn.com/~ramanath/pdf/symptotics-techreport.pdf.

[2] R. Ramanathan, A. Samanta, and T. La Porta. Symptotics: A framework for analyzing the scalability of real-world wireless networks. In *Proceedings of the 9th ACM symposium on PE-WASUN*, pages 31–38. ACM, 2012.

[3] E.N. Ciftcioglu, R. Ramanathan, and T.F. La Porta. Scalability analysis of tactical mobility patterns. In *MILCOM 2013*, pages 1888–1893. IEEE, 2013.

[4] J. Li, C. Blake, D. SJ De Couto, Hu Imm Lee, and R. Morris. Capacity of ad hoc wireless networks. In *Proceedings of the 7th annual international conference on Mobile computing and networking*, pages 61–69. ACM, 2001.

[5] P. Gupta and P.R. Kumar. The capacity of wireless networks. *Information Theory, IEEE Transactions on*, 46(2):388–404, 2000.

[6] A. I Khuri and S. Mukhopadhyay. Response surface methodology. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2(2):128–149, 2010.

[7] H. Tan, M. Chan, W. Xiao, P. Kong, and C. Tham. Information quality aware routing in event-driven sensor networks. In *IEEE INFOCOM, 2010*, pages 1–9, 2010.

[8] Z. M. Charbiwala, S. Zahedi Y. Kim, Y.H. Cho, and M. B. Srivastava. Toward Quality of Information Aware Rate Control forSensor Networks. In *Fourth International Workshop on Feedback Control Implemenation and Design in Computing Systems and Networks*, April 2009.

[9] E.N. Ciftcioglu, A. Yener, and M.J. Neely. Maximizing quality of information from multiple sensor devices: The exploration vs exploitation tradeoff. *Selected Topics in Signal Processing, IEEE Journal of*, 7(5):883–894, Oct 2013.

[10] J. Edwards, A. Bahjat, Y. Jiang, T. Cook, and T.F. La Porta. Quality of information-aware mobile applications. *Pervasive and Mobile Computing*, 11:216–228, 2014.

[11] E. N. Ciftcioglu, A. Michaloliakos, A. Yener, K. Psounis, T. F. La Porta, and R. Govindan. Operational information content sum capacity: From theory to practice. *Computer Networks*, 75:1–17, 2014.

[12] M.Y.S. Uddin, H. Wang, F. Saremi, G.-J. Qi, T. Abdelzaher, and T. Huang. Photonet: a similarity-aware picture delivery service for situation awareness. In *Real-Time Systems Symposium (RTSS), 2011 IEEE 32nd*, pages 317–326. IEEE, 2011.

[13] Y. Jiang, X. Xu, P. Terlecky, T. Abdelzaher, A. Bar-Noy, and R. Govindan. Mediascope: selective on-demand media retrieval from mobile devices. In *Proceedings of the 12th international conference on Information processing in sensor networks*, pages 289–300. ACM, 2013.

[14] A. Bar-Noy, G. Cirincione, R. Govindan, S. Krishnamurthy, T.F. LaPorta, P. Mohapatra, M. Neely, and A. Yener. Quality-of-information aware networking for tactical military networks. In *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2011 IEEE International Conference on*, pages 2–7. IEEE, 2011.

[15] S.A. Chatzichristofis and Y.S. Boutalis. Cedd: color and edge directivity descriptor: a compact descriptor for image indexing and retrieval. In *Computer Vision Systems*, pages 312–322. Springer, 2008.

[16] T.T. Tanimoto. An elementary mathematical theory of classification and prediction. *International Business Machines Corporation*, 1958.

[17] N. Gupta, P.R. Kumar, et al. A performance analysis of the 802.11 wireless lan medium access control. *Communications in Information & Systems*, 3(4):479–304, 2003.