Wireless Netw (2008) 14:465–478
DOI 10.1007/s11276-006-0731-9

Multiconstrained QoS multipath routing in wireless sensor networks

Xiaoxia Huang · Yuguang Fang

Published online: 4 January 2007

Abstract Sensor nodes are densely deployed to accomplish various applications because of the inexpensive cost and small size. Depending on different applications, the traffic in the wireless sensor networks may be mixed with time-sensitive packets and reliability-demanding packets. Therefore, QoS routing is an important issue in wireless sensor networks. Our goal is to provide soft-QoS to different packets as path information is not readily available in wireless networks. In this paper, we utilize the multiple paths between the source and sink pairs for QoS provisioning. Unlike E2E QoS schemes, soft-QoS mapped into links on a path is provided based on local link state information. By the estimation and approximation of path quality, traditional NP-complete QoS problem can be transformed to a modest problem. The idea is to formulate the optimization problem as a probabilistic programming, then based on some approximation technique, we convert it into a deterministic linear programming, which is much easier and convenient to solve. More importantly, the resulting solution is also one to the original probabilistic programming. Simulation results demonstrate the effectiveness of our approach.

This work was supported in part by the U.S. National Science Foundation under grant DBI-0529012, the National Science Foundation Faculty Early Career Development Award under grant ANI-0093241 and the Office of Naval Research under Young Investigator Award N000140210464.

X. Huang · Y. Fang (✉)
Department of Electrical & Computer Engineering,
University of Florida,
Gainesville, FL 32611-6130, USA
e-mail: fang@ece.ufl.edu

X. Huang
e-mail: xiaoxiah@ufl.edu

Keywords Quality of service · Routing · Constrained optimization · Wireless sensor network

1 Introduction

Although small in size, sensor nodes are capable of accomplishing various applications, such as habitat monitoring, telemedicine, surveillance or emergency alarm. Sensor nodes report the sensed information to the sink, which is usually stationary. Depending on different applications, generated packets show diverse attributes. For periodic humidity record packets, as long as it arrives at the processing center or sink, path delay is not critically significant. On the other hand, for multimedia packets, i.e. video packets, if most of them are received in critical time, some loss is acceptable. Another kind of traffic poses strict requirements on both delay and reliability. For example, for a danger warning packet, it should be delivered to the destination as soon as possible without loss. So QoS routing is an important issue in wireless sensor networks.

We investigate both reliability and delay constraints in QoS routing. Here reliability is defined as the packet delivery ratio. Prone to link changes and failures, sensor networks are unreliable. Empirical result from Berkeley [26] shows that the average packet loss ratio increases 5%–10% per link in sensor networks. There are numerous papers on QoS routing. In wired networks, QoS routing with multiple constraints is well-studied. However, unlike wireless networks, reliability is not a key factor in wired networks. Existing literatures in the field of wireless sensor networks focus on a single service metric, such as reliability, delay or energy. Both single path routing and multipath routing have been proposed to solve the problem. However, very few of them consider multiple QoS constraints in sensor networks.

Multiconstrained routing is faced with time complexity and/or space complexity. For wireless networks, complete and accurate state information is not available due to the changing traffic and link quality. Uncertainty makes QoS routing an even tougher problem than in wired networks. Only soft-QoS provisioning is attainable in notoriously unpredictable wireless communications. It is known that finding a path subject to two or more additive constraints is NP-complete [1]. Therefore solving the problem in a heuristic and approximate way is the only reasonable approach for resource-limited sensor nodes. An exciting news from [1] is that topologies leading to an NP-complete behavior of the MCP(Multiple Constraints Problem) problem are less likely to appear, and the worst case complexity of the MCP problem depends on the correlation among the constraint weights.

Delay and reliability need to be satisfied in different ways. Delay is time constrained, yet reliability can be enhanced by path diversity. In this sense, we exploit the time-space efficiency to meet the various characteristics of packets. If a path delay is longer than QoS requirement, then this path is not feasible. In contrast, reliability enhancement is securable through multipath routing. There are two categories of multipath routing [4]. One interpretation is to search multiple paths and choose one of them. The other one is to combine resources of multiple paths for a flow. Our scheme falls into the second category. Many schemes have been proposed to improve reliability based on multipath routing or packet redundancy. Most of them provide heuristic methods without analytical results on the performance. Nevertheless, our routing algorithm design is distinct from them as we formulate the problem in an analytical way. Our goal is to fulfill the soft-QoS requirements in sensor networks. In this context, soft-QoS is defined as guaranteeing the QoS requirements with probability, an approximation of hard-QoS with probability approaching 1. Soft-QoS follows naturally from the inherent random link characteristics of wireless ad hoc and sensor networks. Due to the inherent difficulty of E2E QoS and limited functionality of sensors, some approximate methods have to be applied to deal with the computation complexity problem. In this paper, we first formulate the end-to-end soft-QoS problem as a stochastic programming. Then a distributed routing algorithm is proposed based on the linear programming, which is a deterministic approximate of the end-to-end problem. Our proposed routing algorithm is hop-based, so it is scalable and convenient to implement. As another favorable feature, it circumvents the formidable computation complexity of MCP problem.

The rest of the paper is organized as follows. Section 2 discusses previous work on related topics. Section 3 describes the E2E QoS problem definition and bottlenecks of the problem. Section 4 presents the model and formulation for delay-reliability constrained QoS routing problem. Section 5 illustrates the simulation and discusses the simulation results. Section 6 concludes the paper.

2 Related work

In wired network, many papers have proposed exact or heuristic algorithms targeted at MCP or MCOP(Multi-constrained Optimal Path) problems [1, 3, 8, 11, 12, 14, 16]. Mieghem and Kuipers [3] utilize alternating Dijkstra algorithm to reduce the path search space at the cost of relaxing exactness of solution. Formulating cost as linear combination of additive link weights, Liu and Ramakrushnam prune paths against given constraints at each node [6]. It has been shown that expanding the shortest projected path, via depth-first-search, may result in the needed paths. As a heuristic to MCP, single mixed metric has the drawback of discarding potential useful information, thus generates infeasible path. Thus, in [9], this weakness is relieved by using deviation and average metrics together to reduce the probability of ending up with minimum cost but infeasible paths. In [7], a nonlinear cost function is used to avoid the problem caused by the single mixed metric. A minimum cost path is found when feasible paths exist. Otherwise, the algorithm finds the path minimizing the cost function among infeasible paths. To reduce the complexity of MCP or MCOP problems, many papers resort to approximation methods to shrink the searching space. Yuan trades table size for the reduction of time complexity through quantization of weights at each node and storage of possible optimal paths [8, 10]. Compromising between violation of delay constraint and computation complexity, Orda and Sprintson apply network flow algorithms to attain two disjoint paths as approximation to the optimal feasible paths [25]. In [21], each link has a discrete cost function. Wisely partitioning the delay bound on constituent links of a path, the total cost can be minimized. Based on analysis on the smallest and largest feasible delay at each node with respect to node's processing and buffer capacity, delay assignment strategies are developed to support the end-to-end delay requirement [14]. To address bandwidth and delay constraints in presence of state inaccuracy, Korkmaz and Krunz find the path with the highest probability to meet the constraints [16]. Some approximate algorithms for this problem are presented in [17].

However, wireless sensor networks differ from wired networks in nodes' limited energy, memory and computation capabilities, and link characteristics. A scheme to minimize the cost for delay constrained real-time traffic, while maximize the throughput for non-real-time traffic is proposed in [20]. Chen and Nahrstedt [2] tackle the QoS problem distributively with bounded number of searching paths. Many papers exploit multipath routing to achieve QoS in wireless ad hoc and sensor networks. Based on per-hop channel error rate, which is assumed to be constant across the entire network, the number of outgoing forwarding paths is determined to achieve desired reliability [15]. To combat unreliability, Tsirigos and Hass employ diversity coding and distribute packets over multiple disjoint paths [5]. Gaussian approximation of path success probability, which is tight when the number of paths is sufficiently large, is maximized to reconstruct the original information. Felemban and et al. address both time and reliability constraints in [30]. However, they just use the average link delay and reliability to make routing decisions, so the scheme is not able to adapt quickly for time-varying link conditions. Our paper formulates the problem in a more rigorous way and use both the first and second moments for routing. In [18], Bhatnagar et al. classify paths based on their route lengths. Thus, critical queries go through paths with minimum lengths, and the rest of the traffic is spread uniformly in the network. The algorithm proposed by Das et al. [19] adaptively discovers routes before the occurrence of route errors while transmitting a large volume of data. So it dynamically finds out a series of multiple paths to complete the data transfer. All these papers only consider one QoS constraint. Both disjoint multipath and braided multipath algorithms are explored in [29]. Comparing disjoint multipaths to braided multipaths, braided multipaths have higher resilience to failures with less overhead. Reliability is of great concern in wireless sensor networks due to the fact that sensors are susceptible to failures. Experiments provide some insight into the behavior of link reliability with regard to physical and MAC layers [27]. With combination of frequency based table management, a simple time averaged EWMA estimator [28] is used to model the reliability and achieve reliable routing.

3 E2E QoS multipath routing

Among the two QoS constraints to be explored in this paper, reliability is more difficult to address. Reliability can be characterized by packet delivery ratio, which is defined as the ratio of number of unique packets successfully received by the sink to the number of packets generated by source nodes. For a given path p, the end-to-end reliability can be computed as follows:

$$\prod_{(i,j)\in p} r_{ij}, \tag{1}$$

where r_{ij} is the reliability of link (i, j) on path p. Since reliability is multiplicative, a variation in any one of the link on p would change the end-to-end reliability remarkably. Consider QoS reliability requirement of 95%, if reliability of all outgoing links is below 95% at an intermediate node, there is no feasible path to satisfy the requirement. Even a degradation of 5% on each link will cause a total decrease of 27% on a path p with 6 hops. Also, as the number of hops on the path increases, the E2E reliability decreases. Usually the number of hops in large scale sensor networks is much larger than those in ad hoc networks. So it imposes a severe problem on reliability. For the same p to achieve an E2E reliability of 90%, the geometric mean of reliability of all six links on a six-link path p has to be 98%, which is very restrictive in wireless communications. If the E2E reliability degrades so much that no route can meet the QoS requirement, multipath routing seems to be the only way to enhance the E2E reliability.

3.1 Problem formulation

If at least one route is able to provide the needed QoS requirement, then we could easily obtain a feasible path. However, if some constraint is so aggressive that no single route alone is capable of QoS provision, two different cases are possible. For delay constraint, if a constraint value, say, d_1, which is associated with a data packet, is so restrictive that every path between the source and destination has a delay larger than it, then no path is able to fulfill the delivery of the packet with that constraint. Apparently, there is no feasible path for constraint value d_1. For the other metric, reliability, it is a different case. If there is no single feasible path for a constraint value, say r_1, multipath routing can improve the reliability. Carefully choosing a subset of existing paths, the packet with constraint r_1 can be transferred on all those paths. Although an individual path cannot achieve the performance goal, multiple paths may meet it aggregately. The assembly efficiency of multiple paths is a great boon to unreliable sensor networks. Obviously, there exist many feasible combinations. To save the energy cost, the set with the minimum number of paths is chosen as the forwarding set. We argue that sending a packet on more paths induces more energy cost, because more data packets have to be transmitted. Using more paths introduces more contentions which degrades energy efficiency. Even some paths in the set may have more hops, it is still more energy efficient to confine packets to a few paths. First the question of how to quantify the reliability achieved by a subset of paths needs to be addressed. Then how to choose the energy efficient path set subject to the delay constraint is our main focus. Denote d the sink, which is assumed to be stationary. Let $P(s, d)$ denote the path set of P possible paths from a source node s to d. Each path p_j in $P(s, d)$, $j = 1, 2, \ldots, P$, is associated with delay d_j and reliability r_j. The aggregate reliabilityof multiple paths is approximated as the sum of the reliability of those paths. We formulate the problem as follows:

Problem Definition: $\forall p \in P(s,d)$, at source node s,

$$\text{Minimize} \sum_{j=1}^{P} x_j$$
$$\text{subject to } x_j d_j \le D,$$
$$\mathbf{r} = 1 - \prod_{j=1}^{P} 1 - x_j r_j \ge R,$$
$$x_j = 0 \text{ or } 1, \quad \text{for all} \quad j = 1, 2, \ldots, P$$

where D and R are denoted as the delay and reliability QoS requirements respectively, and $x_j' s$ are decision variables on whether path j is chosen or not. This defines a $0-1$ integer programming problem.

For clarity, notation used in the paper is explained in Table 1.

3.2 Multi-constrained QoS multipath routing under inaccurate path information

The problem definition requires exact information about path quality, which is almost impossible to get in wireless sensor networks. Hence, only soft-QoS provisioning is achievable. Soft-QoS is to provide QoS with certain probability. We can formulate the constraints of the defined problem in a probabilistic way:

$$\text{minimize} \sum_{j=1}^{P} x_j$$

Table 1 Notation

(i, j)	link from node i to node j
$N(i)$	the neighbor set of node i
h_i	hop count from current node i to the sink
r_{ij}	reliability of link l_{ij}
α	soft-QoS probability for delay
β	soft-QoS probability for reliability
L_i^d	hop requirement for delay at node i
L_i^r	hop requirement for reliability at node i
D_i	actual delay of the packet arriving at node i
R_i	reliability requirement assigned to the path through node i
$\mathbf{d}_{ij}$	delay of link l_{ij}, described as a random variable
$\mathbf{r}_{ij}$	reliability of link l_{ij}, described as a random variable
x_j	decision variable of whether link (i, j) is used
d_{ij}	mean of $\mathbf{d}_{ij}$
r_{ij}	mean of $\mathbf{r}_{ij}$
Δ_{ij}^d	standard deviation of $\mathbf{d}_{ij}$
Δ_{ij}^r	standard deviation of $\mathbf{r}_{ij}$

$$\text{subject to } P(x_j \mathbf{d}_j \le D) \ge \alpha, \quad \text{for} \quad D > 0 \tag{2}$$
$$P(\mathbf{r} \ge R) \ge \beta \tag{3}$$
$$x_j = 0 \text{ or } 1, \quad \forall j \in N(i)$$

which is a probabilistic programming, which belongs to stochastic programming. Constraint (3) can be further simplified as

$$P\left(\sum_{j=1}^{P} \log(1 - x_j r_j) \le \log(1 - R)\right) \ge \beta \tag{4}$$

This formulation is a nonlinear programming problem, which could have more than one solution. Solving this nonlinear programming problem at each node once receiving a packet is not practical. So an approximate method, which could significantly simplify the computation of the original problem, while providing comparable fine results, may be more practical. Finding such a practical approximate method is one of our contributions, which will be elaborated in the next few subsections.

3.3 Bottleneck of E2E QoS

Though E2E QoS problem formulated in the previous subsection provides the exact optimal routing solution, it is subject to many inextricable challenges. First, wireless links are susceptible to fading, interference, and traffic variation. Therefore it is almost impossible to obtain the exact instantaneous link state information. So path information, which is accumulated along all links on it, is even more unpredictable. Change of a single link on a path would launch the update of the path information through the network, or network wide flooding on some occasions. Hence, sometimes periodic information exchange mechanism is used to mitigate the effect of inaccurate information. However, frequent information update introduces too much overhead that it may cause congestion and degrade the network performance. There is a tradeoff between the exchange period and accuracy. If the period is long, information may not be precise. On the other hand, if the period is too short, a large amount of overhead is engendered. Second, keeping path metrics consistent at all nodes is an even more formidable problem. Since it takes some time for updates to propagate across the network, some nodes refresh their path information with the received new updates, while other nodes are still using the obsolete information for routing decision. A packet going through nodes with asynchronous path information may miss QoS requirement. Especially for large scale sensor networks, this problem is extremely severe because it is tough to refresh all nodes in a short interval. Third, storage of voluminous E2E path information is dreadfully memory demanding. Possible paths between two nodes may be numerous, whereas a sensor node is equipped with very limited memory. It cannot accommodate all feasible paths. Furthermore, manipulation of E2E information is computationally burdensome for sensor nodes. Delay constrained path problem is known to be NP-hard. The complexity is beyond the computation and energy tolerance of sensors.

Preceding reasons shed light on link based QoS routing. Per hop information is convenient to acquire and maintain at a low overhead cost. The acquired neighbor information is enough to make routing decisions, which saves a large amount of computation. Thus, sensor nodes are free of intricate computation. For those superior features of per hop routing, we propose to approximate path quality based on link quality.

4 Distributed link-based QoS routing

In the problem definition, the problem is formulated based on the end-to-end QoS requirement. The derivation in the former section shows that it is not practical for sensor nodes because of the complexity. So a link-quality based distributed soft-QoS multipath routing which is an approximate of the end-to-end one is to be addressed in this section.

4.1 Requirement partition

Local link metrics and distance to the sink in terms of hop count are used to estimate the path metric. Local link metrics are much easier to acquire and scalable to the network size. By uniformly partitioning current E2E QoS requirements at all downstream hops, we can obtain the hop requirements. If the hop requirement can be achieved at each hop, the end-to-end QoS requirement can also be met. A node can satisfy the hop requirement by selecting next hop nodes based on link conditions. The additive form of delay allows the total available delay to be evenly divided at each hop. On the other hand, the reliability is multiplicative as indicated in (1). Consequently, it takes power form of the requirement. Denote L_i^d and L_i^r as the hop requirements for delay and reliability at node i respectively, h_i as the hop count from node i to the sink, D_i as the actual delay experienced by a packet at node i. As the path from node i to the destination is composed of h_i links, the partitioned requirements at node i can be:

$$L_i^d = \frac{D - D_i}{h_i} \tag{5}$$

$$L_i^r = \sqrt[h_i]{R_i} \tag{6}$$

By introducing D_i and h_i into calculation, the hop requirement for the delay can be adaptively adjusted according to the actual experienced delay over preceding links. Overestimate of delay requirement would tighten the hop delay requirement at downstream nodes, while underestimate would relax the requirement. R is collectively satisfied by several paths, R_i is denoted as the portion of the reliability requirement assigned to the path through node i. R_i is decided by the upstream node of i. As a packet advances towards the sink, h_i at nodes closer to the sink becomes more accurate. So nodes on the route to the sink adaptively adjust the hop requirement. h_i can be easily obtained at the initialization stage, when every node exchanges messages with neighbors to obtain local information. Distribution and determination of the reliability requirement is illustrated in the next section.

4.2 Distribution of reliability requirement

Multiple paths are used as a group to achieve the QoS requirements. Therefore, distribution of the reliability requirement among those paths should first be solved. Nodes determine the reliability distribution on downstream links based on their knowledge. By keeping the reliability distribution on all successor nodes on paths to the sink, the expected reliability can meet the reliability constraint with certain probability. To maintain the reliability assigned by the preceding node, all next hop nodes have to adaptively adjust the reliability distribution among its own successors. As shown in Fig. 1, source node assigns reliability R_1 to its next hop node 1. While neither of the link l_{12} or l_{13} could satisfy this reliability requirement alone. So node 1 distributes reliability requirement R_2 to link l_{12} and R_3 to link l_{13}, so that $1 - (1 - R_2)(1 - R_3) \geq R_1$. The same process is performed at each intermediate node. Finally at sink node d, the three paths, $s \rightarrow 1 \rightarrow 2 \rightarrow 4 \rightarrow d$, $s \rightarrow 1 \rightarrow 3 \rightarrow 5 \rightarrow 7 \rightarrow d$ and $s \rightarrow 1 \rightarrow 3 \rightarrow 6 \rightarrow 7 \rightarrow d$, can achieve the desired reliability additively.

When two nodes share a common successor node, as indicated in Fig. 2, perhaps the successor node receives two copies of a packet asynchronously. Without the knowledge about the other upstream link, the node processes the two identical packets independently and may pick a link twice.

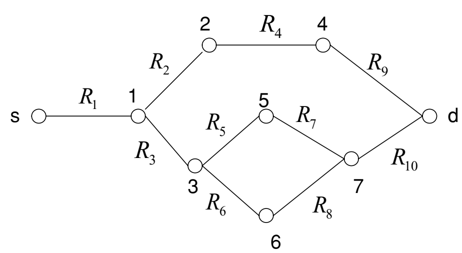


Fig. 1 Reliability distribution between s-d pair

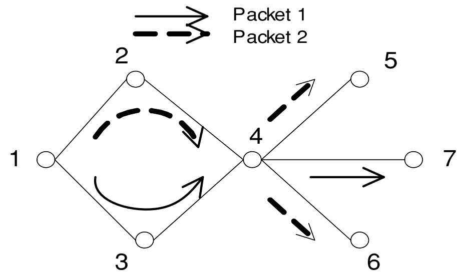


Fig. 2 Common next hop node shared by node 2 and 3

Suppose link l_{47} was selected for packet copy 1 before arrival of packet copy 2. If node 4 chooses links l_{45} and l_{47} for packet copy 2, then the total reliability may decrease as path diversity decreases. So each node should mark the routes it used to forward packets. Once it receives the same packet from a different preceding node, it should select from the unmarked routes to forward the packet. For example, links l_{45} and l_{46} are used to forward packet copy 2. The resulting paths from node 1 are link-disjoint.

4.3 Alternative problem on Hop requirement

In wireless networks, delay and reliability tend to fluctuate with time. To model this phenomenon, we assume that the link delay and reliability are random processes $\mathbf{d}_{ij}(t)$ and $\mathbf{r}_{ij}(t)$. Time index t is omitted for simplicity in the following discussion. We assume that links are independent in terms of delay and reliability. Our goal is to develop a method so that both delay and reliability are assured with high probability while minimizing the number of paths. The more paths participate in communication, the higher potential interference is caused to other flows. This detriments network capacity and energy efficiency. We only employ the first and second moments of delay and reliability in our derivation. Now the new approximate problem to be addressed based on local information is formulated as:

$$\text{minimize} \sum_{j \in N(i)} x_j$$

$$\text{subject to } P\left(x_j \mathbf{d}_{ij} \le L_i^d\right) \ge \alpha, \quad \text{for} \quad L_i^d > 0, \tag{7}$$

$$P\left(\left(1 - \prod_{j \in N(i)} x_j (1 - \mathbf{r}_{ij})\right) \ge L_i^r\right) \ge \beta, \tag{8}$$

$$x_j = 0 \text{ or } 1, \quad \forall j \in N(i)$$

where $x_j's$ are the decision variables, and $\mathbf{d}_{ij}$ and $\mathbf{r}_{ij}$ are the delay and reliability of link l_{ij} at the routing decision instant respectively. This is a probabilistic integer programming. We call it Probabilistic Delay-Reliability Constrained Problem. In the original problem definition, the nonlinear programming is to be solved only at the source based on end-to-end information. In contrast, the approximate problem is to be resolved at all intermediate nodes since the approximate problem is based on hop information. The next two subsections attempt to reduce the computation complexity of the approximation constraints respectively, thus make the approximate solution more appealing.

4.4 Delay constraint linearization

Denote d_{ij} the mean of $\mathbf{d}_{ij}$. Let $(\Delta_{ij}^d)^2$ denote the variance of $\mathbf{d}_{ij}$, as defined in Subsection 4.7. To guarantee that the delay requirement is satisfied with probability no less than α, we must have

$$P\left(x_j \mathbf{d}_{ij} \le L_i^d\right) = P\left(x_j \mathbf{d}_{ij} \le \frac{D - D_i}{h_i}\right) \ge \alpha \tag{9}$$

or

$$P\left(x_j \mathbf{d}_{ij} \ge L_i^d\right) \le 1 - \alpha \tag{10}$$

We estimate the probability according to one-tailed version of Chebyshev's inequality:

$$P\left(\mathbf{X} - m_x \ge a\right) \le \frac{\sigma_x^2}{\sigma_x^2 + a^2}, \quad a > 0$$

which yields:

$$P\left(x_i \mathbf{d}_{ij} \ge L_i^d\right) \le \frac{x_j^2 \left(\Delta_{ij}^d\right)^2}{x_j^2 \left(\Delta_{ij}^d\right)^2 + \left(L_i^d - x_j d_{ij}\right)^2}, \quad L_i^d - d_{ij} > 0 \tag{11}$$

This implies that if d_{ij} satisfies

$$\frac{x_j^2 \left(\Delta_{ij}^d\right)^2}{x_j^2 \left(\Delta_{ij}^d\right)^2 + \left(L_i^d - x_j d_{ij}\right)^2} \le 1 - \alpha$$

then (9) is also valid. Because $x_j = 0$ or 1, $x_j^2 = x_j$. Simplifying the above equation, we obtain

$$x_j \left(\frac{\alpha}{1 - \alpha} \left(\Delta_{ij}^d\right)^2 + 2 L_i^d d_{ij} - d_{ij}^2\right) \le \left(L_i^d\right)^2, \; L_i^d - d_{ij} > 0 \tag{12}$$

As the deterministic estimate for (9), (12) is linear and solvable. Note that Chebyshev bound is rather loose, so the solution space of (12) is smaller than the original one. Tuning α to an appropriate value to relax the solution space, could get feasible solutions to the original problem. So we add a non-negative multiplicative factor $\omega < 1$ to it. $\alpha' = \omega\alpha$ is used in computation in (12). However, there is a tradeoff between the solution space and precision. A small α would include a larger solution space, at the risk of increasing the probability of expired packets. To achieve the best performance, this tuning parameter has to be carefully chosen.

Recall that we assume no knowledge about current link delay, except the first and second moment. This is a pessimistic estimation, which usually can be improved in real networks. There is a coherence time period in which a link keeps relatively static. Abrupt changes of link delay do not happen frequently. Unfortunately, link condition is affected by so many dynamic factors, such as fading, interference, contention, traffic flow and topology, that no accurate model has been developed to describe it so far. Since there is no available link model, link estimation is an alternative. If MCMP algorithm performs well under this worst case condition, some prediction measures could be incorporated to further improve its efficiency.

4.5 Reliability constraint linearization

In our problem, link reliability is an additive constraint, whereas delay is a bottleneck constraint, which is determined by the minimal one. Therefore, reliability is more complicated to deal with than delay. The current measured value of reliability r_{ij} is the time average of all finished transmissions. Assume that link reliability, $\mathbf{r}_{ij}$, is a random process with mean and variance r_{ij} and $(\Delta^r_{ij})^2$ without specific p.d.f. Inequality (8) is a nonlinear constraint, which is unsolvable for capability restricted sensors. Simplifying the constraint to a linear function is more advantageous and practical. The original problem is reduced to selecting a set of paths meeting the partitioned reliability requirement at every time instant. Observe that the total reliability requirement can be achieved by multiple links

$$L^r_i = \sqrt[h_i]{R_i} = 1 - \prod_{j \in N(i)} (1 - x_j R_{ij}) \tag{13}$$

With this formula, the link reliability requirement can be easily obtained. But there may exist more than one solution without regard to feasibility. So we add some constraints to restrict the solutions to the feasible ones.

Then the reliability requirement is satisfied if

$$\begin{aligned} P\big(\mathbf{r} \geq L^r_i\big) &= P\Big(\bigcup_{j \in N(i)} x_j \mathbf{r}_{ij} \geq x_j R_{ij}\Big) \\ &= \prod_{j \in N(i)} P(x_j \mathbf{r}_{ij} \geq x_j R_{ij}) \end{aligned}$$

The equation holds because the reliability of each link is independent. Then we can obtain the constraint

$$P\big(\mathbf{r} \geq L^r_i\big) = \prod_{j \in N(i)} P\left(x_j \mathbf{r}_{ij} \geq x_j R_{ij}\right) \geq \beta \tag{14}$$

Denote $E[\mathbf{r}_{ij}] = \mu(\mathbf{r}_{ij}) = r_{ij}$, $\sigma^2_{ij} = (\Delta^r_{ij})^2$. Let $\mathbf{r}'_{ij}$ be the sum of all previous transmissions over link l_{ij}. The pth transmission either succeeds or fails, so it's reasonable to model a single transmission as a Bernoulli trial ξ_p with a finite variance, whose realization is either 1 or 0. However, without the assumption that all transmissions over a link are identically distributed, i.e. all transmissions have diverse success probabilities, the transmission process is not a Bernoulli process. Assume all transmissions are independent, then

$$\mathbf{r}'_{ij} = \sum_{p=1}^{M} \xi_p = M \frac{\sum_{p=1}^{M} \xi_p}{M} = M\mathbf{r}_{ij} \tag{15}$$

where M is the number of transmissions over link l_{ij}. Note that $\mathbf{r}'_{ij}$ is not a binomial distribution because each ξ_p has different success probability. As M goes large, $\mathbf{r}'_{ij}$ is approximately Gaussian distributed according to the Central Limit Theorem, $\mathbf{r}'_{ij} \sim N(Mr_{ij}, M^2(\Delta^r_{ij})^2)$.

Take the logarithm on both hands of the inequality (14),

$$\sum_{j \in N(i)} \log(P(x_j \mathbf{r}_{ij} \geq x_j R_{ij})) \geq \log \beta \tag{16}$$

Observe that

$$\log P(x_j \mathbf{r}_{ij} \geq x_j R_{ij})) = \begin{cases} \log 1 = 0, & \text{when } x_j = 0 \\ \log(P(\mathbf{r}_{ij} \geq R_{ij})), & \text{when } x_j = 1 \end{cases}$$

So we can rewrite (16) as

$$\sum_{j \in N(i)} x_j \log(P(M\mathbf{r}_{ij} \geq MR_{ij})) \geq \log \beta \tag{17}$$

As $M\mathbf{r}_{ij} = \mathbf{r}'_{ij}$ is Gaussian distributed, we have

$$P(\mathbf{r}'_{ij} \geq MR_{ij}) = Q\left(\frac{MR_{ij} - Mr_{ij}}{M\Delta^r_{ij}}\right) \tag{18}$$

Substitute (18) into (17)

$$\sum_{j \in N(i)} x_j \log\left(Q\left(\frac{R_{ij} - r_{ij}}{\Delta_{ij}}\right)\right) \geq \log \beta \tag{19}$$

where

$$\prod_{j\in N(i)} (1 - x_j R_{ij}) \le 1 - L_i^r \tag{20}$$

Again, take logarithm on both sides of (20)

$$\sum_{j\in N(i)} x_j \log(1 - R_{ij}) \le \log\left(1 - L_i^r\right)$$

Notice that the Q-function in (19) is actually a constant, hence the constraint is linear. Now both inequalities are linear.

These two inequalities form deterministic linear constraints for reliability. Combined with (12), the path selection problem as a deterministic linear programming is formulated as follows:

Problem Formulation: At each node i,

$$\text{minimize} \sum_{j\in N(i)} x_j$$

subject to

$$x_j\left(\frac{\alpha}{1-\alpha}\left(\Delta_{ij}^d\right)^2 + 2L_i^d d_{ij} - d_{ij}^2\right) \le {L_i^d}^2, \quad \text{when } L_i^d - d_{ij} > 0 \tag{21}$$

$$\sum_{j\in N(i)} x_j \log\left(Q\left(\frac{R_{ij} - r_{ij}}{\Delta_{ij}}\right)\right) \ge \log\beta, \tag{22}$$

$$\sum_{j\in N(i)} x_j \log(1 - R_{ij}) \le \log\left(1 - L_i^r\right) \tag{23}$$

$$x_j = 0 \text{ or } 1, \quad \forall\, j \in N(i)$$

$$0 \le R_{ij} \le r_{ij}, \quad \forall\, j \in N(i)$$

The new optimization problem is a deterministic estimate of the problem formulated in (8). There are many efficient algorithms to solve this integer programming(IP) [23]. The number of constraint is $2|N(i)| + 2$ and the number of decision variables is $|N(i)|$. Since the IP is solved locally at each intermediate node, the problem size is relatively small. Apparently, the size of the problem is proportional to the node density.

4.6 Loop avoidance

Since every node lacks the global knowledge about the network, the routing algorithm may engender loops. A packet may induce a large delay on the loop until link state changes to break the loop. In the worst case, a packet may never arrive at the sink because a node on the loop would discard the packet if it expires. For a packet with loose reliability requirement, it may just be sent on a single path, which is seriously affected by a loop. On the other hand, for a packet transferred on multiple paths, reliability will degrade if some copies of the packet are lost due to loops. Therefore, overcoming loops is indispensable to secure the effectiveness of our routing algorithm.

Lemma 1. *Assume that a packet is being transferred on a path $p = n_1, n_2, \ldots, n_q$, let $h(\cdot)$ denote the number of hops from a node to the sink. If $h(n_{u+1}) < h(n_u)$, for all $n_u \in p$, then there is no loop.*

The minimum distance to the sink of each node, in terms of hop count, is included in the neighbor table to eliminate loops. Before making routing decisions based on link state, a node checks the minimum distances of all neighbors. It only opts for neighbors with fewer hop counts to the sink as eligible successor nodes. With this confining condition, our routing algorithm becomes greedy through advancing one hop towards the sink at every successful transmission. Applying this result, using hop count effectually prevents loops and produces equal length paths. For a node with distance h, it can only choose nodes with valid distance of $h - 1$, as its forwarding nodes. This assures that the even partition of QoS requirements is exact and consistent in terms of distance. A node discovers the minimum distance to the sink at the stage of exchanging neighbor information.

4.7 Calculation of Δ_{ij}^d and Δ_{ij}^r

Adaptively values of Δ_{ij}^d and Δ_{ij}^r may provide better estimates of path performance than the fixed ones due to the dynamic link conditions. A simple method is to determine current $\Delta_{ij}^d(t)$ and $\Delta_{ij}^r(t)$ based on previous values of $d_{ij}(t-1)$, $r_{ij}(t-1)$, and $\Delta_{ij}^r(t-1)$, and current values of $d_{ij}(t)$ and $r_{ij}(t)$. In real wireless networks, the link delay and reliability at successive time instants are correlated in time. The variances of the two constraints ought to embody time correlation in link quality. Therefore, our estimation mimics RTT estimation for timer management in TCP.

$$\Delta_{ij}^d(t) = (1-\rho)\Delta_{ij}^d(t-1) + \rho|d_{ij}(t) - d_{ij}(t-1)| \tag{24}$$

$$\Delta_{ij}^r(t) = (1-\gamma)\Delta_{ij}^r(t-1) + \gamma|r_{ij}(t) - r_{ij}(t-1)| \tag{25}$$

Tunable forgetting parameter ρ and γ smooths the variations of d_{ij} and r_{ij} in time. For realistic wireless sensor networks, this is reasonable because current link state depends on historical link state.

Delay and reliability are calculated at each node based on the previous transmissions and receptions. Delay is obtained

Table 2 MultiConstrained MultiPath (MCMP) routing algorithm

0. *Delay-reliability Constrained Multipath Routing Algorithm*
1. $candidate = \{l_{ij} | h_j < h_i, j \in N(i)\}$;
2. $forwarding = \emptyset$;
3. $L_i^d = \frac{D-D_i}{h_i}$;
4. if $(L_i^d \leq 0)$
5. discard the packet and return;
6. else
7. $L_i^r = \sqrt[h_i]{R_i}$;
8. Update $\Delta_{ij}^d(t)$ and $\Delta_{ij}^r(t)$ using equations (24) and (25);
9. while $(candidate \neq \emptyset)\{$
10. if (inequality (21) holds for d_{ij} and $\Delta_{ij}^d(t)$)$\{$
11. add link l_{ij} to *forwarding;*
12. $candidate = candidate - l_{ij}\}\}$;
13. Applying the branch and bound algorithm to solve the reliability constraint in the Problem Formulation in the reduced solution space given in *candidate*.

from round trip time, and reliability is the average successful communications. Those information can be acquired by observing MAC layer handshakes, i.e. RTS-CTS-DATA-ACK in IEEE 802.11, which does not impose additional control overhead.

4.8 Algorithm for MCMP

Based on the design goal, optimization can be towards different objectives. The objective function could be minimum number of selected paths to minimize energy consumption, minimax reliability, maximin average delay.

Our goal is to utilize the multiple paths to augment network performance with moderate energy cost. Thereupon, the objective function is to minimize the number of paths, as indicated in the Problem Formulation.

There are many existing algorithms [23] which can be applied to solve our linear integer programming. Table 2 outlines an efficient algorithm, called MCMP, to solve our problem.

Lines 1 to 7 initialize values to be used in the following computation. Line 8 to 12 check the eligibility of each link and decide the forwarding sets. As described in Section 4.4, they check the feasibility of links with delay constraint. Line 13 solves the optimization problem constrained by the reliability constraint in the solution space obtained in the preceding steps.

5 Simulation

We conducted extensive simulations to evaluate the performance of MCMP algorithm. As mentioned before, our simulations assess the worst case performance as a benchmark. By worst case, we mean that link delay and reliability always change suddenly at any transmission instant and are not predictable. Our interest is to examine the feasibility of our approximate method of probabilistic modeling of unknown link delay and reliability in wireless sensor networks. If it achieves good performance in this general case, it will work for well-defined models too. It has the potential to achieve better performance by implementing some prediction measures to track the link condition. Even we use adaptive standard deviation and mean of link state in MCMP algorithm, they are not powerful enough under volatile link condition. Comparison is conducted with single path routing(SP), braided multipath routing [29] and God routing. God routing is defined as the routing algorithm that each node is aware of the instantaneous link delay and reliability, and selects multiple paths based on the exact knowledge, which is usually not available in reality. God routing serves as an ideal routing algorithm, thus its performance is the upper bound that is attainable by multipath routing. The closeness to God routing presents the efficacy of MCMP algorithm. The single path routing just selects an individual path, if any, which can fulfill the QoS requirement. In braided multipath routing, multiple paths are discovered at the path establish stage. Sink chooses the best path as primary path, others as alternate paths when the primary one fails. In our simulation, the best path is the one with the shortest delay between the source and sink pair. The performance difference between MCMP and single path routing or braided multipath routing reflects the performance improvement gained through MCMP.

5.1 Simulation setup

The simulation is implemented in PARSEC [31], which provides the parallel discrete-event simulation capability. The simulations are performed on a uniform topology consisting of 50 nodes spread in a square area of 100 m × 100 m. Sink is at the top left of the field. The transmission range of all nodes is 25 m. Success probability of each transmission is randomly picked from [0.8, 1], which implies that the link reliability ranges from 0.8 to 1. Link delay is also randomly distributed in the range of [1, 50] ms. The link delay is the elapsed time for successfully transmitting a packet after receiving it. So it includes queuing time, contention time, transmission time, retransmission time and propagation time. As MCMP does not assume and utilize the distribution of link delay for routing decision, it can be applied to network with any link delay distribution. Here we choose uniform distribution following the popular simulation or numerical models in [2] [5] and [16]. Link states randomly vary at all transmission instants. So it is a worst case comparing to real networks. The delay requirement is uniformly distributed between 120 to 260 ms with an interval of 10 ms, which produces 15 delay requirement levels. Likewise, the reliability requirement uniformly ranges from 0.7 to 1 with an interval of 0.05. This gives 7 distinct reliability requirement levels. Each simulation run randomly selects ten nodes to generate packets at the speed of 1 packet/second. Data packet has a fixed size of 150 bytes. Each simulation runs for 60 minutes. We change the random seed to generate different traffic across the network at each of the 12 runs. For the same traffic setting, three algorithms are executed for comparison.

5.2 Performance metrics

Evaluated performance metrics include on-time packet delivery ratio, packet delivery ratio, expiration ratio, and average packet delay. On-time packet delivery ratio is the number of packets successfully received satisfying the QoS requirement to the total number of generated packets. Packet delivery ratio is the ratio of the number of packets successfully received to the total number of the generated packets. Since we have two QoS constraints, the packet delivery ratio is explored against each one. Successful reception has different definition for the two constraints. For delay requirement, a packet of successful reception is the packet received satisfying delay requirement. While for reliability requirement, a packet of successful reception is defined as the packet arriving at the sink node without loss. So on-time packet delivery ratio exhibits the performance of packets with different delay requirements. While the packet delivery ratio demonstrates the performance of packets with different reliability requirements. Expiration ratio is the ratio of the number of packets that arrived at the sink violating the delay requirements to the total number of generated packets. The average packet delay is the average end-to-end delay experienced by successfully received packets. We investigate the performance metrics against delay and reliability requirements separately.

Because Chebyshev's inequality is fairly loose, α is tuned so that MCMP behaves better under strict constraints. This parameter implies the relationship between the on-time packet delivery ratio and expiration ratio. Smaller α relaxes the time constraint, hence the solution space includes more routing candidates. As a result, the on-time packet delivery ratio increases, and expiration ratio increases too. Larger α reduces the solution space, so the miss ratio increases. Consequently, the packet delivery ratio decreases, and the expiration ratio decreases too.

5.3 Simulation results

The following figures show the simulation results, in which both the probability of delay and reliability constraint α and β in (22) are set to 95%. To display the relationship between performance metrics and QoS requirements, figures are shown separately with respect to delay and reliability. Results demonstrate that MCMP outperforms single path routing remarkably, and approaches approximately 95% of which for God routing. MCMP also achieves better performance than braided multipath routing. Because braided multipath routing is more applicable to network with relatively static link condition than constantly changing link condition. Another reason is that braided multipath routing only use one path for data delivery. On the contrary, MCMP simultaneously uses multiple paths to diminish the impairment of link dynamics.

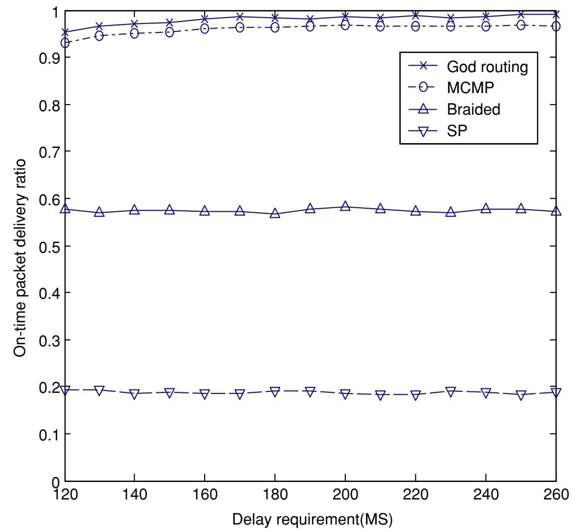


Fig. 3 On-time packet delivery ratio vs. delay requirement

Figure 3 illustrates that packets with slack delay requirements have a higher on-time delivery ratio. The superior on-time delivery ratio of MCMP over single path routing and braided multipath routing validates the potentness of multipath routing. Owing to enhanced reliability, much more packets are received successfully at the sink node. MCMP improves performance by more than 50% over braided multipath routing. Without precise knowledge of link delay and reliability, MCMP has a slightly lower on-time delivery ratio than God routing.

Figure 4 indicates the average end-to-end delay of successfully received packets. As God Routing has full knowledge

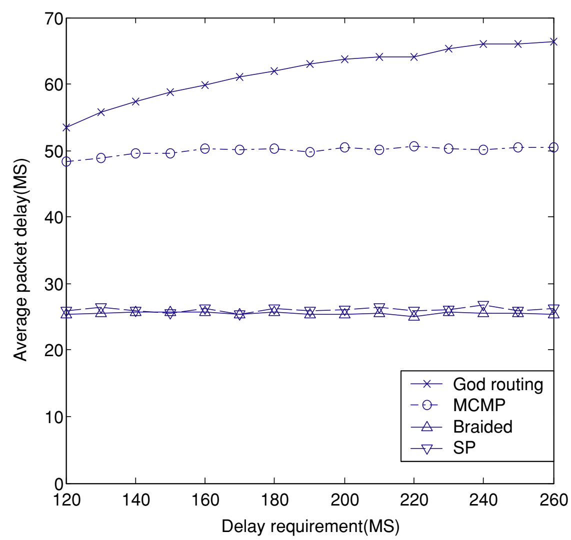


Fig. 4 Average End-to-End packet delay vs. delay requirement

of link states, it spans delay better than MCMP. Although the tuning parameter is used to confine the expiring ratio to a relatively small value, the algorithm is still a little conservative in estimating end-to-end delay. Consequently, delay is restricted in a small range compared to God routing. This explains that some packets are dropped at intermediate nodes, resulting in a lower packet delivery ratio compared to God routing. Hence, there is a tradeoff between the expiring ratio and on-time delivery ratio. Single path routing drops most of the packets generated farther away from the sink and has the smallest delay among three algorithms. Only packets originated in a few hop distance to the sink can be received. Braided multipath routing also has the least end-to-end delay because it uses the shortest paths.

Figure 5 manifests the reliability performance of packets with different reliability requirements. The packet delivery ratio is almost the same for all reliability requirements, because they achieve the highest reliability constrained by α and β. The reliability performance of God routing also confirms this. As implied by God routing, the achievable reliability is around 99%. MCMP attains the reliability around 96% with small expiring ratio. Hence, MCMP achieves 95% of the delivery ratio as we set it as β. Due to the relatively low link reliability, single path routing has to drop most packets due to multihop paths. Although braided multipath routing uses several alternate paths as backup to recover from packet loss, it just uses one path to transfer packets, so the end-to-end reliability is inferior compared to MCMP. Simulation results show that our MCMP algorithm prominently enhances the QoS routing performance without accurate link condition information. Although MCMP loses some of the packets due to its conservative partial solution space, it still approaches the best performance excluding God routing. In reality, link conditions may persist for a short period of time instead of changing sharply at all time, as assumed in our simulations. Therefore, it is expected to perform better in real networks than our simulation result.

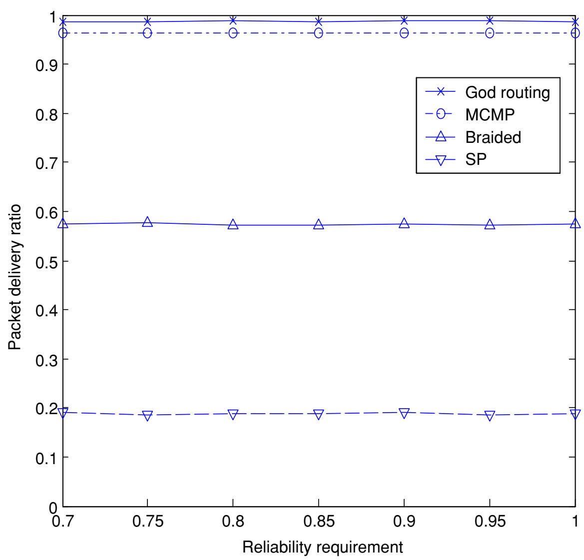


Fig. 5 Average packet delivery ratio vs. reliability requirement

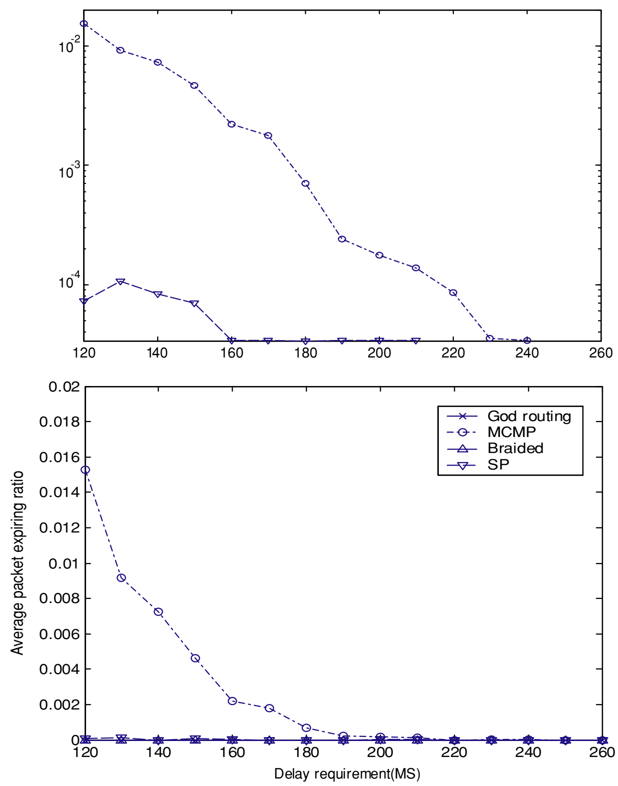


Fig. 6 Average packet expiring ratio vs. delay requirement when link reliability is between [0.8, 1]

Figure 6 exhibits the distribution of expiring ratio. To show the detail of the lower graph, the upper part shows the expiring ratio in log scale to get a better view of it. Part of curves corresponding to MCMP and single path routing disappears in the upper graph because the corresponding points are of value zero. Braided multipath routing and God routing achieve zero expiring ratio, so they are not visible in the upper graph. Note that the average expiring ratio is lower than $1 - \alpha = 5\%$, because some packets are lost or discarded before arriving at the sink. Expiring ratio of MCMP drops as delay requirement increases, due to the same reason as on-time packet delivery ratio vs. delay requirement. For delay requirements above 180 ms, the expiring ratio is negligible, as they have a large solution space. Single path routing and braided multipath routing have minute expiring ratios because most packets from distant nodes have been discarded or lost before arriving at the sink. For single path routing, the majority of received packets are from nodes within a short distance to the sink, so delay requirement is easy to satisfy. It achieves zero expiring ratio when the delay requirement is longer than 160 ms.

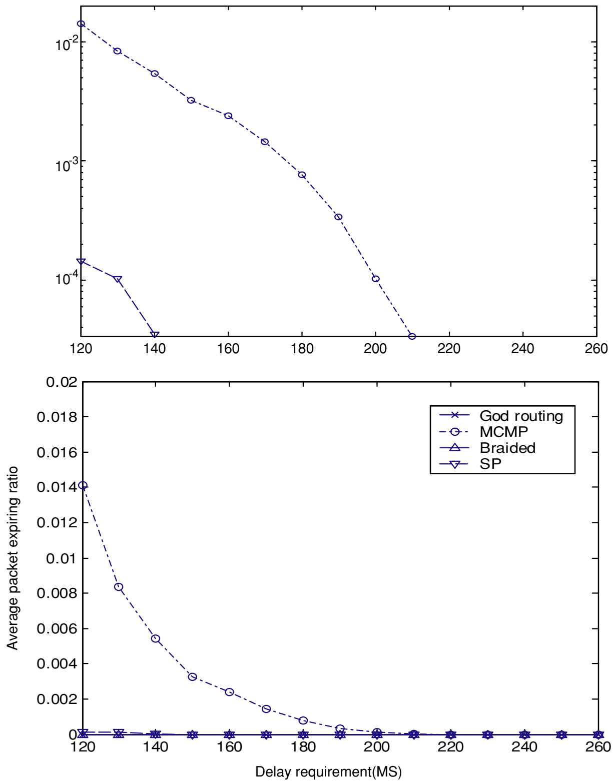


Fig. 7 Average packet expiring ratio vs. delay requirement when link reliability is between [0.9, 1]

There is another set of simulations with the same setting as the previous one except that the link reliability varies between [0.9, 1]. As expected, the performance is improved, though slightly. More reliable links result in less uncertainty in the deterministic estimate. Therefore, it is a better approximate of the stochastic programming for E2E QoS routing, and improves the performance. From Fig. 10, it can be seen that reliability increases from 96% to 99%, achieving almost the same performance as God routing. On the other hand, as delay requirement remains the same, in Figs. 8 and 9, on-time packet delivery ratio and the end-to-end packet delay do not change significantly. Higher link reliability ensures more packets to be delivered to the sink on time, so the expiring ratio decreases in Fig. 7. To have a close view of expiring ratio in the lower graph, the upper part shows the expiring ratio in log scale. Again, disappeared curves and partial curves are of value zero. MCMP achieves zero expiring ratio for packets with delay requirement greater than 210 ms. When the delay requirement is beyond 140 ms, all other routing algorithms achieve zero expiring ratio. MCMP which is insensitive to link reliability in contrast to the single path routing and braided multipath routing, shows that higher link reliability does not affect the performance significantly. It follows naturally that MCMP algorithm is more suitable for unreliable wireless sensor networks. The transmission cost confirms this too as in Fig. 11. Figure 11 shows

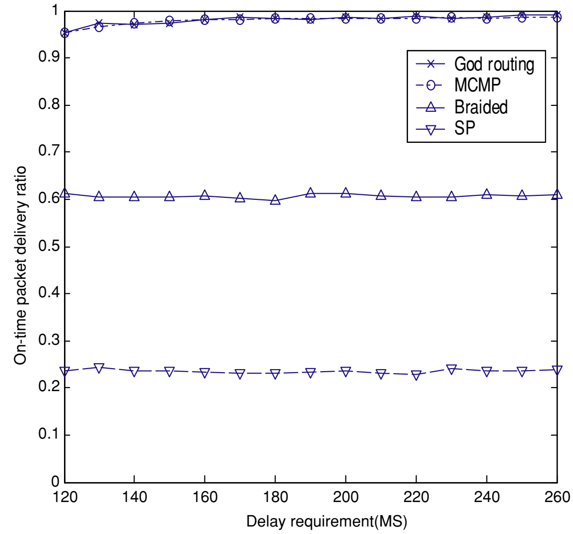


Fig. 8 On-time packet delivery ratio vs. delay requirement

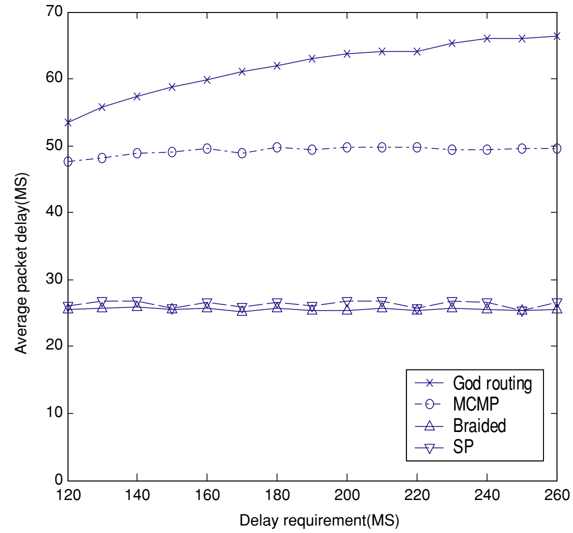


Fig. 9 Average End-to-End packet delay vs. delay requirement

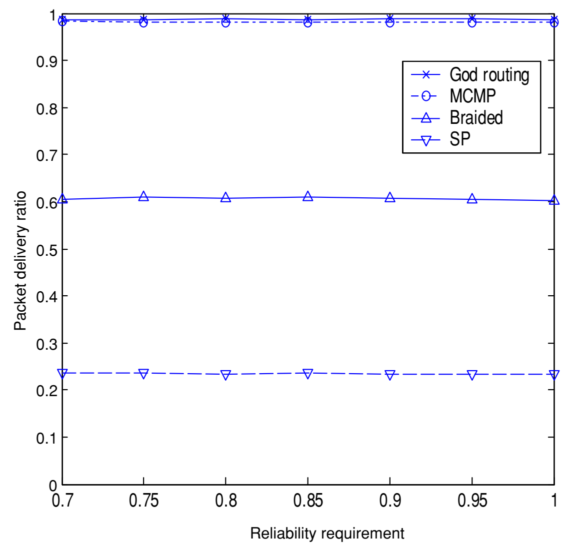


Fig. 10 Average packet delivery ratio vs. reliability requirement

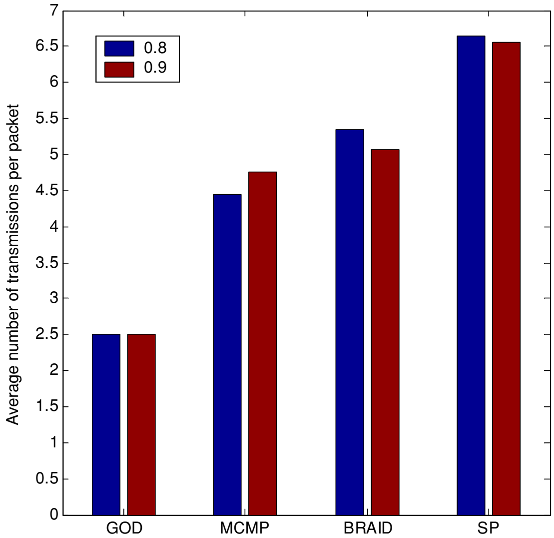


Fig. 11 Comparison of transmission cost per packet

that MCMP achieves lower transmission cost than braided multipath routing and single path routing, but is inferior to God routing. The number of transmissions per packet actually reflects the energy efficiency. Since the more transmissions needed per packet, the higher energy consumption for delivering a packet. MCMP gains more advantage when the link reliability is 0.8. The interesting phenomenon is as the link reliability increases from 0.8 to 0.9, the number of transmissions per packet drops for braided multipath routing and single path routing, but increases for MCMP. The reason is that when the link reliability is relatively low, less redundant packets are delivered to the sink. So MCMP achieves better energy efficiency when the link is relatively unreliable. As the link reliability increases, the packet delivery ratio improves slightly but more duplicate packets successfully go through multiple paths, causing higher redundancy.

6 Conclusion and future work

In this paper, we proposed a probabilistic modelling of link state for wireless sensor networks. Based on this model, an approximation of local multipath routing algorithm is explored to provide soft-QoS under multiple constraints, such as delay and reliability. The existing routing algorithms for sensor networks just consider one constraint, through single path, multipath routing or flooding. Inherent computation complexity and prohibitive overhead associated with multi-constrained QoS routing problem pose serious challenges. Our MCMP routing algorithm trades precise link information for sustainable computation, memory and overhead for resource limited sensor nodes. Simulation results validate our scheme as its performance achieves near optimal performance achieved by the multipath routing with perfect link knowledge. Though some feasible paths are excluded from solution space, the approximation algorithm still yields impressive outcome.

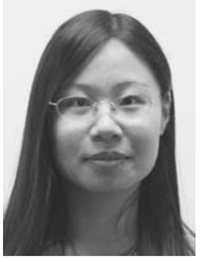
The rather pessimistic model of link state in the current work provides the first step research. Some estimation technique can be used to strengthen the robustness of MCMP. The accuracy of the estimation determines the tightness of the performance of MCMP to the theoretical upper bound. Also, employing some coding scheme could reduce the redundancy. We are currently working on this to further improve the performance of MCMP. Moreover, MCMP is not significantly affected by mobility because it uses local information only. It has the flexibility to deal with topology changes caused by node movement without maintaining global routing information.

Acknowledgments This work was supported in part by the U.S. National Science Foundation under grant DBI-0529012, the National Science Foundation Faculty Early Career Development Award under grant ANI-0093241 and the Office of Naval Research under Young Investigator Award N000140210464.

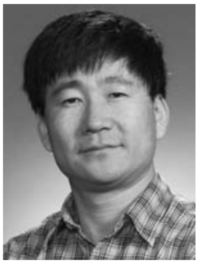
References

1. P.V. Mieghem and F.A. Kuipers, On the complexity of QoS routing, Computer Communications 26(4) (2003) 376–387.
2. S. Chen and K. Nahrstedt, Distributed quality-of-service routing in ad hoc networks, IEEE Journal on Selected Areas in Communications 17(8) (August 1999).
3. P.V. Mieghem and F.A. Kuipers, Concepts of exact QoS routing algorithms, IEEE/ACM Trans. on Networking 12(5) (October 2004) 851–864.
4. S. Chen and K. Nahrstedt, An overview of quality-of-service routing for the next generation high speed networks: problems and solutions, IEEE Network, Special Issue on Transmission and Distribution of Digital Video 12(6) (November/December 1998) 64–79.
5. A. Tsirigos and Z.J. Hass, Analysis of multipath routing, part 2: mitigation of the effects of frequently changing network topologies, IEEE Trans. on Wireless Communications 3(2) (March 2004) 500–511.
6. G. Liu and K.G. Ramakrushnam, A*Prune: an algorithm for finding K shortest paths subject to multiple constraints, IEEE INFOCOM 2001 2 (April 2001) 743–749.
7. T. Korkmaz and M. Krunz, Multi-constrained optimal path selection, IEEE INFOCOM 2001 2 (April 2001) 834–843.
8. X. Yuan, Heuristic algorithms for multiconstrained quality-of-service routing, IEEE/ACM Trans. on Networking 10(2) (April 2002) 244–256.
9. P. Khadivi, S. Samavi, T.D. Todd and H. Saidi, Multi-constraint QoS routing using a new single mixed metric, 2004 IEEE Intl. Conference on Communications 4 (June 2004) 2042–2046.
10. W. Xiao, Y. Luo, B.H. Soong, A. Xu, C.L. Law and K.V. Ling, An efficient heuristic algorithm for multi-constrained path problems, Vehicular Technology Conference 2002 3 (September 2002) 24–28.
11. J.M. Jaffe, Algorithms for finding paths with multiple constraints, Networks 14 (Spring 1984) 95–116.
12. H.D. Neve and P.V. Mieghem, A multiple quality of service routing algorithm for PNNI, in: *Proceedings Of the ATM workshop IEEE* (May 1998) pp. 324–328.

13. D. Bhatnagar, B. Deb and B. Nath, Service differentiation in sensor networks, Fourth International Symposium on Wireless Personal Multimedia Communications (September 2001).
14. T.F. Znati and R. Melhem, Node delay assignment strategies to support end-to-end requirement in heterogeneous networks, IEEE/ACM Trans. on Networking 12(5) (October 2004) 879–892.
15. B. Deb, S. Bhatnagar and B. Nath, ReInForM: Reliable information forwarding using multiple paths in sensor networks, Local Computer Networks 2003, In: *Proceedings. 28th Annual IEEE International Conference on* (October 2003) pp. 406–415.
16. T. Korkmaz and M. Krunz, Bandwidth-delay constrained path selection under inaccurate state information, IEEE/ACM Trans. on Networking 11(3) (June 2003) 384–398.
17. Y. Xiao and K. Thulasiraman, Approximation and heuristic algorithms for delay constrained path selection under inaccurate state information, In: *Proceedings of the First Intl. Conf. on Quality of Service in Heterogenenous Wired/Wireless Networks(QSHINE'04)* (October 2004) pp. 130–137.
18. S. Bhatnagar, B. Deb and B. Nath, Service differentiation in sensor networks, Fourth Intl. Symposium on Wireless Personal Multimedia Communications (September 2001).
19. S.R. Das, A. Mukherjee, B. Bandyopadhyay, K. Paul and D. Saha, Improving quality-of-service in ad hoc wireless networks with adaptive multi-path routing, IEEE Globecom 2000 1 (November 2000) 261–265.
20. K. Akkaya and M. Younis, An energy-aware QoS routing protocol for wireless sensor networks, Distributed Computing Systems Workshops in: *Proceedings. 23rd Intl. Conference on* (May 2003) pp. 710–715.
21. D. Raz and Y. Shavitt, Optimal partition of QoS requirements with discrete cost functions, IEEE Journal on selected areas in communications 18(12) (December 2000) 2593–2602.
22. S. Wang, D. Xuan, R. Bettati and W. Zhao, Providing absolute differentiated services for real-time applications in static-priority scheduling networks, IEEE/ACM Trans. on Networking 12(2) (April 2004) 326–339.
23. M.S. Bazaraa, J.J. Jarvis and H.D. Sherali, Linear Programming and Network Flows, Third Edition (December 2004).
24. J.R. Birge and F. Louveaux, Introduction to stochastic programming (1997).
25. A. Orda and A. Sprintson, Efficient algorithm for computing disjoint QoS paths, IEEE INFOCOM 2004 1 (March 2004) 727–738.
26. N. Ota, D. Hooks, P. Wright, D. Ausiander and T. Peffer, Poster abstract: wireless sensor networks characterization-application to demand response energy pricing, ACM SenSys'03 (November 2003) pp. 334–335.
27. J. Zhao and R. Govindan, Understanding packet delivery performance in dense wireless sensor networks, ACM SenSys'03 (November 2003) pp. 1–13.
28. A. Woo, T. Tong and D. Culler, Taming the underlying challenges of reliable multihop routing in sensor networks, ACM SenSys'03 (November 2003) pp. 14–27.
29. D. Ganesan, R. Govindan, S. Shenker and D. Estrin, Highly-resilient, energy-efficient multipath routing in wireless sensor networks, Mobile Computing and Communications Review(MC2R) 1(2) (2002).
30. E. Felemban, C.G. Lee, R. Boder and S. Vural, Probabilistic QoS guarantee in reliability and timeliness domains in wireless sensor networks, Proceedings of IEEE INFOCOM 2005 (March 2005).
31. http://pcl.cs.ucla.edu/projects/parsec/.



Xiaoxia Huang received her BS and MS in the Electrical Engineering from Huazhong University of Science and Technology in 2000 and 2002, respectively. She is completing her Ph.D. degree in the Department of Electrical and Computer Engineering at the University of Florida. Her research interests include mobile computing, QoS and routing in wireless ad hoc networks and wireless sensor networks.



Yuguang Fang received a Ph.D. degree in Systems Engineering from Case Western Reserve University in January 1994 and a Ph.D degree in Electrical Engineering from Boston University in May 1997. He was an assistant professor in the Department of Electrical and Computer Engineering at New Jersey Institute of Technology from July 1998 to May 2000. He then joined the Department of Electrical and Computer Engineering at University of Florida in May 2000 as an assistant professor, got an early promotion to an associate professor with tenure in August 2003 and to a full professor in August 2005. He holds a University of Florida Research Foundation (UFRF) Professorship from 2006 to 2009. He has published over 200 papers in refereed professional journals and conferences. He received the National Science Foundation Faculty Early Career Award in 2001 and the Office of Naval Research Young Investigator Award in 2002. He has served on several editorial boards of technical journals including IEEE Transactions on Communications, IEEE Transactions on Wireless Communications, IEEE Transactions on Mobile Computing and ACM Wireless Networks. He have also been activitely participating in professional conference organizations such as serving as The Steering Committee Co-Chair for QShine, the Technical Program Vice-Chair for IEEE INFOCOM'2005, Technical Program Symposium Co-Chair for IEEE Globecom'2004, and a member of Technical Program Committee for IEEE INFOCOM (1998, 2000, 2003–2007).