M-Plan: Multipath Planning based Transmissions for IoT Multimedia Sensing

[1]Min Chen, [2]Junfeng Wang, [3]Kai Lin, [4]Di Wu, [5]Jiafu Wan, [6]Limei Peng, [7]*Chan-Hyun Youn
[1] School of Computer Science and Technology, Huazhong University of Science and Technology, China
[2] School of Information Engineering, Zhengzhou University,China
[3] School of Computer Science and Technology, Dalian University of Technology, China
[4] School of Information Science and Technology, Sun Yat-sen University, China
[5] School of Mechanical & Automotive Engineering, South China University of Technology, China
[6] Dept. of Industrial and Information System Engineering, Ajou University, South Korea
[7] School of Electrical Engineering, KAIST, Daejeon, South Korea
*Corresponding author: Chan-Hyun Youn

***Abstract*—Multimedia transmissions for IoT (Internet-of-Things) sensing has a high demand of route capacity and tight requirements of end-to-end delay. In this paper, we address the problems on how to guarantee delay-related QoS requirements and to balance the energy consumption, while using multipath routing to offer high transmission capability for IoT multimedia sensing. This motivates us to design a Multipath Planning for Single-Source based transmissions routing scheme, namely MPSS, which establishes desirable multiple route paths following B-spline trajectories based on geographical information of source and sink node, sending and receiving angles, and inter-path distance. We further utilize a factor of hop distance to reduce the cumulated error of each hop due to the density of nodes, and to guarantee the delay-related QoS requirements. A Multipath Planning for Multi-Source routing scheme is also designed, namely MPMS, to assign the angle scope according to the source node's priority and traffic. Experimental results show that MPSS can effectively generate well-patterned multiple spline-based routes, and the end-to-end delay is under control according to the delay QoS requirement, while the total energy consumption is minimized.**

I. Introduction

Motivated by recent hardware advances, some powerful sensor nodes can be equipped with multimedia functionalities that enable them to sever as wireless multimedia sensor networks (WMSNs). Compared to traditional WSNs with the intrinsic feature of scalar data collection, WMSNs are more beneficial to elaborate some complicated events and phenomena, yet raise new challenging issues in QoS provisioning for multimedia streaming, as well as optimally reduce energy consumption. Since the compressed video bit stream is extremely sensitive to transmission errors, error control techniques such as forward error correction (FEC) and automatic repeat request (ARQ) are necessary to obtain the high reliability required by video services.

The general multi-path routing is considered to be a single source to send data to the sink node [1]. In this paper, a multipath planning for Single-source based transmissions routing scheme (MPSS) is proposed to utilize spline trajectory for multipath routing while guaranteeing the end-to-end delay and minimizing the energy consumption. Furthermore, we propose a Multipath Planning for Multi-Source routing scheme (MPMS). The main contributions of this paper include:

- In MPSS, we utilize the flexibility of spline to establish disjointed route with well-defined pattern while the inter-path interference is reduced.
- We generalize theoretical models of energy consumption and end-to-end delay to balance the energy and delay by estimating the desired hop count of each path based on its spline length, which converts the energy and delay tradeoff to a simple metric, hop count.
- Given the desired hop count, MPSS optimally assigns the transmission range for each sensor with power control capability so that the energy consumption is reduced but the end-to-end delay is still guaranteed.
- We also design MPMS, which is able to allocate path according to the source node's priority and the traffic flow. Source node is able to join and exit dynamically.

The rest of the paper is organized as follows. Section II presents the energy and delay models, and our motivation to design MPSS. The details of delay and energy models, spline controlled multipath routing, MPSS and MPMS algorithms are presented in Section III. Finally, we show the results of the performance evaluation of MPSS and MPMS in Section IV. At last we conclude our work in Section V.

II. Preliminaries and Overview

A. MPSS architecture overview

Most of the traditional routing schemes intend to maximize the routing progress by introducing maximized hop distance to minimize the end-to-end delay. But when the application specifies a delay QoS boundary, earlier arrivals of packets are practically meaningless [2]. In the case that sensor nodes have power control capability [3], sensors do not have to always utilize maximized power to achieve maximum transmission range. One of the key points in MPSS design is to adjust the transmission power to a smaller level, but still to guarantee the required end-to-end delay of the paths.

The idea of energy-delay trade-off is motivated by an interesting feature of many new sensor devices which have the capability to transmit at different power levels [4].

1) Delay model: The hop delay (denoted by T_{hop}) for any node can be simply estimated as a constant. We estimate T_{hop} based on 802.11b Distributed Contention Function (DCF) standard and modeling work. The transmission delay for a successful data delivery includes data transmission delay, T_{DATA}, and acknowledgement delay, T_{ACK}.

Given the QoS delay constraint, T_{QoS}, the required hop count to guarantee QoS delay, H_{QoS}, can be estimated as

$$H_{QoS} = \left\lfloor \frac{T_{QoS}}{T_{hop}} \right\rfloor. \tag{1}$$

2) Energy model: We adopt a general energy model used in [6], in which the components of energy consumption include: *Idle energy consumption*, *Transmission energy consumption*, *Receiving energy consumption*, and *Control signaling energy consumption*. We generalize the energy model of one hop transmission as:

$$E_{hop} = C \cdot D^{\alpha} \quad \rightarrow \quad E_{hop} = D^2 \tag{2}$$

where C is a constant value, D is the adjusted hop distance, and α is the path loss exponent depending on the environment, typically 2 when free space propagation is assumed. For the sake of simplicity, we set C to 1, and α to 2. Therefore, a packet transmission through the kth path (denoted by P_k) from the source to the sink will consume energy E_k as

$$E_k = \sum_{i=1}^{|P_k|} \left(D_{n_{i-1}n_i}\right)^2 \tag{3}$$

where $D_{n_{i-1}n_i}$ is the transmission distance from n_{i-1} to n_i, n_0 is the source n_s and $n_{|P_k|}$ is the sink n_t.

B. MPSS architecture statement

The MPSS scheme consists of two components: **1)Calculation of hop count**: based on the models of hop delay and application-specific delay boundary, we estimate the required hop count for transmission. **2)Energy saving by power control**: any node will adjust the transmission power for an optimized transmission range calculated based on the path length and hop count.

The definition of MPSS problem is stated as follows.

Defination 1: Given an application-specific QoS delay, T_{QoS}, and the number of desired paths, K, the goal is to generate K route paths between a source and a sink, while the end-to-end delay is controlled to be very close to T_{QoS} and energy consumption for a success date delivery along each path E_k is to be minimized.

Let D_{st} denote the distance of the midline/shortest path. To obtain the adjustable scope of T_{QoS}, we first calculate its lower bound as, $T_{QoS} \geq \frac{D_{st}}{R_{max}}$, which is achieved when data is transmitted along the midline path with maximum transmission range (i.e., R_{max}). Otherwise, there is no path which can satisfy the QoS delay requirement. On the other hand, T_{QoS} also cannot be set to be too large inefficiently, such that we have the following theorem.

Theorem 1: for a certain network topology, an multimedia application is allowed to adjust application-specific end-to-end delay T_{QoS} subject to the following constraint:

$$\frac{C.D_{st}}{R_{\max}}.T_{hop} \geq T_{QoS} \geq \frac{D_{st}}{R_{\max}}.T_{hop} \tag{4}$$

where C is typically set to 2 in this paper.

According to the delay model in Section II, T_{hop} can be considered as a constant. In order to guarantee T_{QoS}, the expected end-to-end hop count H_{QoS} is calculated by Eq. 1.

Theorem 2: Let R^* denote the lower bound of the average hop distance along a path. In order to guarantee that the delay is lower than T_{QoS} when the midline path is utilized to transmit data, R^* is given by

$$R^* \geq \frac{D_{st}}{H_{QoS}} \tag{5}$$

When constructing multiple paths, P_k, $k = 1,2,...,K$, let R_k denote the strategic transmission range by power control. Then, the adjustable scope of R_k is given by

$$R^* \leq R_k \leq R_{max} \tag{6}$$

where R_{max} is the maximum transmission range.

If H_{QoS} hop count is used in the K paths, the QoS delay can be satisfied for these multiple paths. According to *Definition 1*, the problem now is to find the optimal R_k for each path P_k, $k = 1,2,...,K$.

Theorem 3: Let L_k denote the length of P_k. The optimal R_k for each node along P_k is given by

$$R_k = \frac{L_k}{H_{QoS}} \tag{7}$$

where $R^* \cdot H_{QoS} \leq L_k \leq R_{max} \cdot H_{QoS}$ according to Eqns. 6,7. Let $L_{max} = R_{max} \cdot H_{QoS}$ denote the maximum path length.

We prove the minimization of energy consumption by assigning R_k to average hop distance at P_k as follows:

$$\begin{aligned} \underset{\{D_{n_{i-1}n_i}\}}{\text{minimize}} \quad & E_k = \sum_{i=1}^{H_{QoS}} \left(D_{n_{i-1}n_i}\right)^2 \\ \text{subject to} \quad & \sum_{i=1}^{H_{QoS}} D_{n_{i-1}n_i} \approx L_k. \end{aligned}$$

We solve the problem by utilizing the Cauchy-Schwarz Inequality which is

$$\begin{aligned} \sum_{i=1}^{H_{QoS}} \left(D_{n_{i-1}n_i}\right)^2 = \frac{\sum_{i=1}^{H_{QoS}} \left(D_{n_{i-1}n_i}\right)^2 \cdot \sum_{i=1}^{H_{QoS}} 1}{H_{QoS}} \geq \\ \frac{\left(\sum_{i=1}^{H_{QoS}} D_{n_{i-1}n_i} \cdot 1\right)^2}{H_{QoS}} \approx \frac{L_k^2}{H_{QoS}}, \end{aligned} \tag{8}$$

if and only if $D_{n_0n_1} = D_{n_1n_2} = ... = D_{n_{H_{QoS}-1}n_{H_{QoS}}} = \frac{\sum_{i=1}^{H_{QoS}} D_{n_{i-1}n_i}}{H} \approx \frac{L_k}{H_{QoS}} = R_k$, E_k gets the minimized value. So we equally use $R_k = \frac{L_k}{H_{QoS}}$ for each hop of the path P_k to minimize the energy consumption.

In this manner, a series of K paths with application-specific end-to-end delay T_{QoS} can be produced.

III. M-Plan for Single-source based Streaming

We decompose the MPSS design functionality into the following four components, i.e., splines trajectory generating approach, directional controlled algorithm, spline based next-hop-selection mechanism and accuracy enhancement algorithm for power control.

A. Splines trajectory generating approach

Some researchers have already used spline trajectory for geographical routing [1], [5]–[7] , in which they utilized the advantages of spline-controlled routing as follows:

- A spline can be controlled by partial parametric functions, so that intermediate nodes can calculate the next hop. The source can control the direction and pattern of paths due to the high flexibility of splines.
- The route length of a spline-controlled path can be calculated by the integral of the parametric functions of the curve.

1) Single path generating approach: A spline based path in MPSS consists of 3 contiguous *cubic B-Spline* segments as shown in Fig. 1. Each segment is plotted by an individual parametric function, which is determined by four control points, such that the parametric function of spreading segment, $S_{sp}(t)$, is controlled by n'_s, n_s, n_{cs} and n_{ct}; the parametric function of spreading segment, $S_{pl}(t)$, is controlled by n_s, n_{cs}, n_{ct} and n_t, and the parametric function of spreading segment, $S_{cv}(t)$, is controlled by n_{cs}, n_{ct}, n_t and n'_t. They can be calculated as follows:

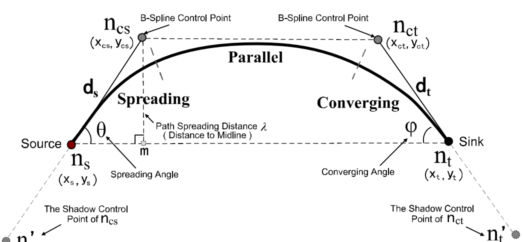


Fig. 1. How spline controls a route path

$$\begin{cases} S_{sp}(t) = \frac{1}{6}n_s'(1-t)^3 + \frac{1}{6}n_s\left(3t^3-6t^2+4\right) \\ \quad + \frac{1}{6}n_{cs}\left(-3t^3+3t^2+3t+1\right) + \frac{1}{6}n_{ct}t^3 \\ S_{pl}(t) = \frac{1}{6}n_s(1-t)^3 + \frac{1}{6}n_{cs}\left(3t^3-6t^2+4\right) \\ \quad + \frac{1}{6}n_{ct}\left(-3t^3+3t^2+3t+1\right) + \frac{1}{6}n_t t^3 \\ S_{cv}(t) = \frac{1}{6}n_{cs}(1-t)^3 + \frac{1}{6}n_{ct}\left(3t^3-6t^2+4\right) \\ \quad + \frac{1}{6}n_t\left(-3t^3+3t^2+3t+1\right) + \frac{1}{6}n_t' t^3 \end{cases} \tag{9}$$

where t is in the range of 0 to 1 for each segment. The absolute coordinates of n_{cs}, (x_{cs}, y_{cs}), is given by:

$$\begin{cases} x_{cs} = x_s + d_s \cdot \cos(\theta + \gamma) \\ y_{cs} = y_s + d_s \cdot \sin(\theta + \gamma) \end{cases} \tag{10}$$

where (x_s, y_s) is the absolute coordinates of n_s, $\gamma = \tan^{-1}\frac{y_t - y_s}{x_t - x_s}$ is the angle from the midline $\overrightarrow{n_s n_t}$ to the global X axis, θ is the spreading angle (i.e.,$\angle n_t n_s n_{cs}$), and d_s is the spreading control distance (i.e., $\overrightarrow{n_s n_{cs}}$) which is calculated as follows.

$$d^s = \frac{\lambda}{\sin(\theta)} \tag{11}$$

where λ is the deviation distance to midline.

Similarly, the absolute coordinates of n_{ct}, (x_{ct}, y_{ct}), can be obtained if converging angle φ (i.e., $\angle n_s n_t n_{ct}$), and the spreading control distance d_t (i.e., $\overrightarrow{n_t n_{ct}}$) are given. In this paper, φ is equal to θ while d_t is the same as d_s.

Based on the coordinates of n_{cs} and n_{ct}, the other two control points, n_s' and n_t', can be easily calculated by:

$$n_s\prime = 2n_{cs} - n_s, \quad n_t\prime = 2n_{ct} - n_t \tag{12}$$

which guarantees the spline start from n_s and terminate at n_t.

2) From single path to multiple paths: According to Eqns. 9, 10, 11, 12, a spline based path can be determined by (x_s, y_s), (x_t, y_t), θ and λ. Note that (x_s, y_s) and (x_t, y_t) are the same for all paths between the source and the sink. The source can transmit a series of packets each specifying a different combination of θ and λ,

Let K denote the number of paths; θ_k denote the spreading angle of the kth path (i.e., P_k), and λ_k denote the deviation distance to midline of P_k, where $k = 1, 2, 3, ...K$. Then, a series of θ_k can be calculated as follows:

$$\theta_k = \Delta\theta \cdot |k - 1 - \frac{(K-1)}{2}| \tag{13}$$

where $\Delta\theta$ is the inter-path spreading angle. The best approach is to spatially distribute multiple paths as evenly as possible. Thus, the deviation angle between two adjacent paths is set to a same value (i.e, $\Delta\theta$).

And a series of λ_k can be calculated as follows:

$$\lambda_k = \Delta\lambda \cdot |k - 1 - \frac{(K-1)}{2}| \tag{14}$$

where $\Delta\lambda$ is the inter-path distance. In order to reduce the interference between packets transmitted over two adjacent paths, $\Delta\lambda$ is typically set to R_{max}.

Given the path number K, $\Delta\theta$ and $\Delta\lambda$ are two important control knobs for the multipath construction in MPSS. $\Delta\theta$ adjusts the density of multiple paths in proximity of the source node, while $\Delta\lambda$ controls how "fat" the spline should go or how far should the parallel segment of the spline deviate from the midline.

3) Calculating the inter-path spreading angle: Let θ_{max} denote the spreading angle of the path with maximum path. In MPSS, $\Delta\theta$ is calculated as follows.

$$\Delta\theta = \frac{2\theta_{max}}{K-1} \tag{15}$$

As mentioned in Section II-B, MPSS will first obtain the hop count H_{QoS} according to the required QoS delay T_{QoS}, then calculate the length of the longest path, i.e. $L_{max} = R_{max} \cdot H_{QoS}$, which determines the value of θ_{max}.

B. Directional controlled algorithm

In MPSS, the path directions are controlled by path spreading angle.

1) Calculating spline length: In Fig. **??**, let n_i represent the current node; n_{i+1} represent the practical next hop node of n_i; v_i and v_{i+1} denote the virtual nodes of n_i and n_{i+1}, respectively. Let (x_t, y_t) represent the coordinates of v_i. Let ΔL_t be the length increment at v_i. Let $(x_{t+\Delta t}, y_{t+\Delta t})$ denote the coordinates of the next node in the spline which is a step of length increment far from v_i. ΔL_t is calculated by:

$$\Delta L_t = \sqrt{(x_{t+\Delta t} - x_t)^2 + (y_{t+\Delta t} - y_t)^2} \tag{16}$$

The next length increment at node $(x_{t+\Delta t}, y_{t+\Delta t})$ can be calculated by

$$\Delta L_{t+\Delta t} = \sqrt{(x_{t+2\Delta t} - x_{t+\Delta t})^2 + (y_{t+2\Delta t} - y_{t+\Delta t})^2} \tag{17}$$

and so on. The total length of one segment of the spline is approximated as the summation of all the length increments. For example, the length of spreading segment, L^{sp}, can be approximated as follows.

$$L^{sp} = \sum_{j=0}^{\lfloor \frac{1}{\Delta t} \rfloor} \Delta L^{sp}_{j \cdot \Delta t}, \tag{18}$$

where $\lfloor \frac{1}{\Delta t} \rfloor$ is the number of increments when varying t from 0 to 1. The smaller Δt is, more accurate spline length can be obtained while increasing the computational complexity. The length of parallel segment, L^{pl}, and the length of converging segment, L^{cv} are calculated in a similar manner. Finally, the spline length, L, is equal to $L^{sp} + L^{pl} + L^{cv}$.

Since $\Delta L^{sp}_{j \cdot \Delta t}$, $\Delta L^{sp}_{j \cdot \Delta t}$ and $\Delta L^{sp}_{j \cdot \Delta t}$, $j \in [0, ..., \lfloor \frac{1}{\Delta t} \rfloor]$, are mainly determined by the key control parameters θ and λ, we denote $L = GetLength(\theta, \lambda)$ as the integral approximation function for calculating spline length.

2) Calculating spreading angle: Given a path length, the calculation of spreading angle is the inverse function of $L = GetLength(\theta, \lambda)$. In this section, we propose an exhaustive algorithm to obtain the value of the spreading angle given a certain path length L with an acceptable error boundary, e, as shown in Algorithm 1.

Algorithm 1 Algorithm to get spreading angle

```
begin
notation
  τ is a small increment of θ
  Ł is specified path length
initialization
  θ = 0;
  Ł' = GetLengh(θ,λ);
while |Ł'-Ł| > e do
    θ = θ + τ;
    Ł' = GetLengh(θ,λ);
end while
  Return θ;
```

Given L_{max}, θ_{max} can be obtained based on Algorithm 1. Consequently, the inter-path spreading angle $\Delta\theta$ can be calculated according to Eq. 15.

C. Spline based next-hop-selection mechanism

The the location of virtual next hop node means the ideal location of current node's next hop. $(x_{t+\Delta t}, y_{t+\Delta t})$ represents the location of the node which is a step of increment far from v_i, then let $(x_{t+j\Delta t}, y_{t+j\Delta t})$ denote the location of the node which is j steps farther than v_i. When j is increased, $(x_{t+j\Delta t}, y_{t+j\Delta t})$ gradually approaches to the boundary of the forwarding area of current node n_i. Given the example shown in Fig. **??**, v_{i+1} is the virtual next-hop node of n_i, and v_{i+2} is the virtual next-hop node of n_{i+2}. MPSS will then select as the next hop node whose distance is closest to the location of the virtual next-hop node, instead of the neighbor closest to the sink as in traditional geographic routing protocols. In Fig. **??**, n_{i+1} is the closest neighbor to v_{i+1} within the transmission range of n_i, and thus selected as the next hop of n_i. n_{i+1} will continue to find its next hop, and so forth, until the sink node is reached. As a result, a source-sink path will be established.

D. Accuracy enhancement algorithm for power control

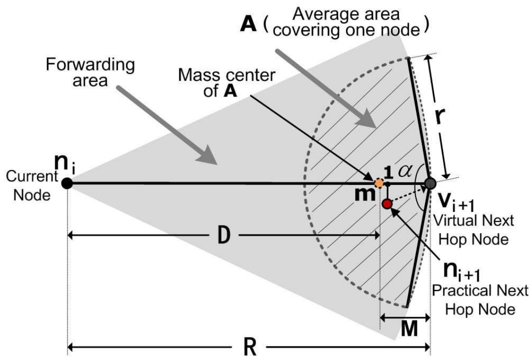


Fig. 2. Calculation of ρ

In most of the cases, the practical next hop is nearer to the current node than the virtual next hop, which incurs an accumulative error. Therefore, we introduce an enhancement factor, ρ, to enlarge the hop distance a little longer than the strategic transmission range R_k (see Eq. 6) in order to eliminate the accumulative error.

Regardless of the distribution of nodes, the mean number of nodes within a unit area is expected to be unchanged statistically. In Fig. 2, v_{i+1} is the virtual next hop node of the current node n_i. Within the forwarding area, the practical next hop node, n_{i+1}, is the neighbor closest to v_{i+1}. Let P_i^{i+1} denote the projected progress, which is the distance between the current node n_i and the projection point (i.e., point 1 in Fig. 2) of n_i. Though the transmission range of R is used by n_i, the effective progress towards the sink, P_i^{i+1}, is smaller than R in most cases. Let m represent the mass center of A. From the statistic point of view, P_i^{i+1} can be approximated as the distance between n_i and m, which is denoted by D.

In order to obtain the location of m, we need to determine the pie area whose size is equal to A and center is located at v_{i+1}. Let r denote the radius of the pie area. Given A and R, r and α can be approximated by solving the following equations set:

$$\begin{cases} \cos\frac{\alpha}{2} = \frac{r}{R} \\ A \approx \frac{\alpha}{2\pi} \cdot \pi r^2 \end{cases} \tag{19}$$

Then, D is equal to $R - M = R - \frac{4}{3} \cdot \frac{r}{\alpha} \cdot \sin^2\frac{\alpha}{2}$. Finally, the ρ factor is calculated by:

$$\rho = \frac{R}{D} = \frac{R}{R - \frac{4}{3} \cdot \frac{r}{\alpha} \cdot \sin^2\frac{\alpha}{2}} \tag{20}$$

E. M-Plan for Multi-source based Streaming

Considering sink is located in the center of the simulated scenario, multiple source nodes send data to sink by multipath individually. When designing the protocol of MPSS,the following problems should be considered. Firstly, how to assign the angle scope to each source node when source nodes are unevenly distributed around sink. Secondly, how to assign angle to each source node when source nodes have different weight.Thirdly, how to handle the situation where the group of source nodes expand and shrink dynamically. In order to reduce the interference of the radio signal, we assign each source node with a specific angle scope according to the weight of the source node and makes the paths disjointed.

The angle's distribution algorithm is based on the weight of the source node, i.e., $Q_i = P_i * F_i$, where P_i is the priority of the ith node, and F_i represents the data flow of node i. Then, the total weight of all source nodes can be calculated as $Q_{total} = \sum_{i=1}^{n} Q_i$. Based on Q_{total}, the angle scope of node i can be obtained: $Angle_i = \frac{Q_i}{Q_{total}} \times 360$.

The source's angle scope can be calculated by the angle scope allocated by sink. Fig 3 shows the angle scope of three source nodes(Src1,Src2,Src3).

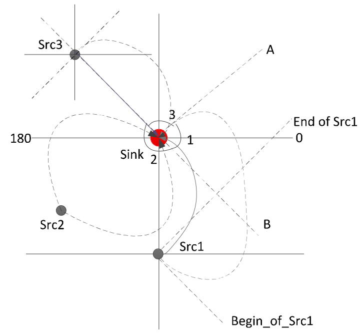


Fig. 3. Angle assign of Source

IV. Performance Evaluation of MPSS And MPMS

A. Simulation Methodology and Parameters

We implement our protocols and perform simulations using OPNET Modeler. The network with 2000 nodes is randomly deployed over a 2000m $\times$ 1000m field. We let one sink stay at a corner of the field and one source node be located at the diagonal corner. R_{max} is set to 60m. The data rate of the wireless channel is 1 Mb/s, and all data packets have 1KB payload. We assign each path a traffic load of 100Kbps. The Δt in Eq. 18 is set to 0.01 and the path number is set to 5. We first carry out simulation to verify energy and delay models. Next, we show how MPSS generates B-spline based multiple paths. Then, we compare pure spline multipath routing to MPSS schemes. Finally, MPSS is compared with a representative multipath routing protocol (i.e., DGR [1]).

B. Verification of delay and energy models

The spreading angle θ is varied from 40^o to 120^o, and the maximum transmission power is used in all the intermediate sensor nodes. When θ is increased, L_e and H_{ete} are increased. However, the average hop delay keeps nearly constant around the average value 0.00264s. And we also notice that the practical average hop distance is around 53m which is smaller than R_{max} of 60m. It is because the distance progress of a real next hop is smaller than that of the corresponding virtual next hop, which motivates the introduction of the enhancement factor as illustrated in Section III-D.

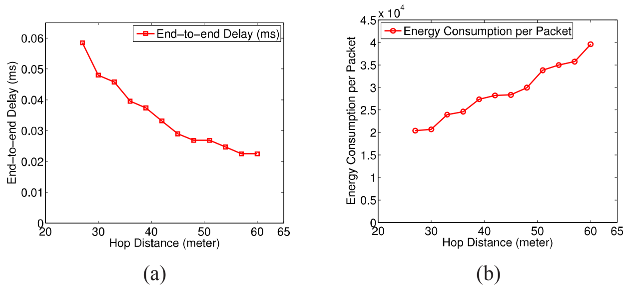


Fig. 4. Energy and delay tradeoffs in MPSS. (a) Impact of R_k on energy. (b) Impact of R_k on delay.

To verify the impact of R_k on MPSS in terms of energy and delay, we vary R_k from 25m to 60m and carry out simulation studies. It can be observed in Fig. 4 that energy and delay exhibit a tradeoff relation with regard to R_k. Given a certain delay requirement, it is desirable to adjust R_k to a value as low as possible to minimize energy consumption while satisfying the delay requirement.

C. Performance comparison of MPSS

In this section, the delay requirement T_{QoS} is set to 0.03s, and the path number is set to five. Then, the three schemes, i.e., Pure spline multipath scheme, MPSS without ρ and MPSS with ρ, are evaluated and compared.

- *Pure spline multipath scheme*: it has no functionality to guarantee QoS delay. As show in Fig. 5(a), most of the paths have various delays ranging from 0.0027s to 0.039s. Among the five paths, two of them can not achieve the delay requirement. On the other hand, the maximum transmission range is always used, and the path length is not controlled as well. Thus, the average energy consumption in Fig. 5(b) is the highest among all the schemes.
- *MPSS without ρ*: it roughly make end-to-end delay near to the boundary of required delay in Fig. 5(c), but a bit larger due to the accumulative difference error at each hop from the virtual next hop to the real next hop. Compared to pure spline multipath scheme, the energy consumption is reduced as shown in Fig. 5(d).
- *MPSS with ρ*: The calculation of ρ is done by the source node when assigning R_k to path P_k. By looking at values of ρ in in Table I, we observe that ρ is inversely proportional to the transmission range. MPSS achieved higher accuracy to approach the virtual next hop at each hop of each path. So, the end-to-end delay is guaranteed while more energy is saved due to the reduced hop count. From the simulation results in Fig. 5(e) and Fig. 5(f), the end-to-end delays are exactly bounded at the edge of T_{QoS}, and the energy consumption per packet is relatively smaller than MPSS without ρ enhancement.

D. Performance comparison with varying delay objectives

We further evaluate MPSS with and without ρ enhancement compared with DGR [1], K-shortest [7], and pure spline-based multipath routing to investigate how MPSS outperforms others in guaranteeing end-to-end delay and energy saving by following test items:

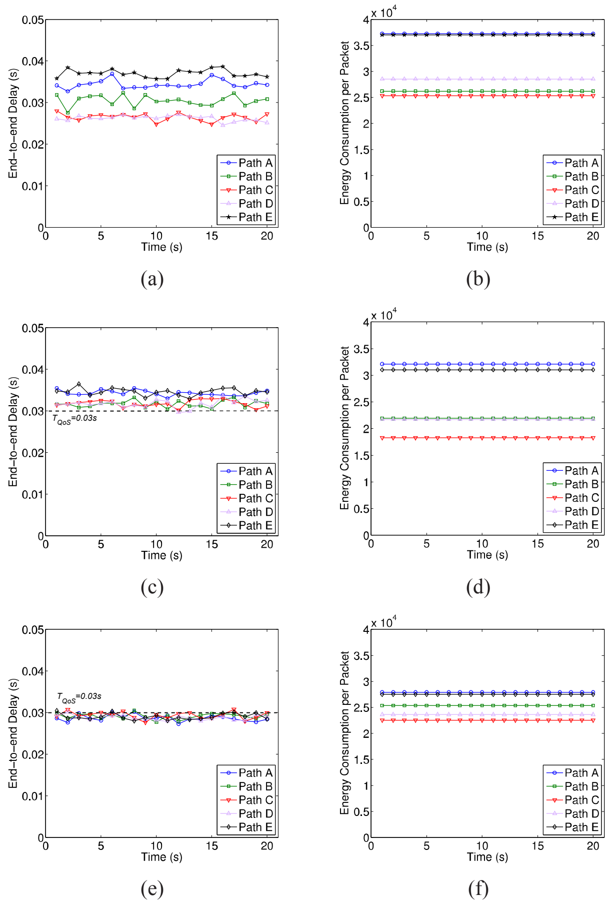


Fig. 5. Performance comparison of spline based schemes: (a) delay in pure spline multipath scheme. (b) energy in pure spline multipath scheme. (c) delay in MPSS without ρ. (d) energy in MPSS without ρ. (e) delay in MPSS with ρ. (f) energy in MPSS with ρ.

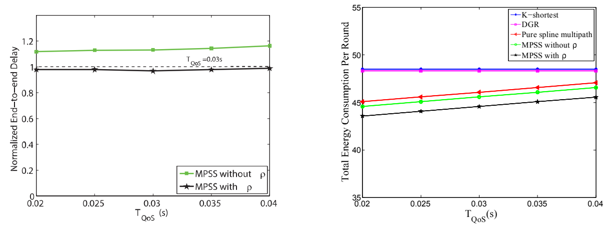
TABLE I
CALCULATION ABOUT ρ

R	r	α	M	ρ
20	13.55171	1.225163	4.875808	1.322385
30	12.88951	1.354283	4.98198	1.199136
40	12.62171	1.412363	5.017767	1.143438
50	12.47526	1.445718	5.035071	1.111978
60	12.38265	1.467424	5.045079	1.091804

- *Average End-to-end Delay*: denoted by T_{ete}, which includes all the delays during transmissions.
- *Energy Consumption per Round*: denoted by E, which is the total energy consumption of all paths to successfully transmit one packet to the sink.

To verify the impact of T_{QoS} on difference schemes, we vary T_{QoS} from 0.02s to 0.04s and carry out extensive simulation studies. Fig. 6(a) shows the comparison of MPSS with and without ρ in terms of the normalized delay. The curve of MPSS with ρ is almost identical with the ideal line, and thus achieving a better performance to approach QoS delay requirement.

Fig. 6(b) shows the comparison of energy performance. The energy of K-shortest, DGR routing schemes remain constant, because changing T_{QoS} has no effect on them. Among all of the scheme, MPSS with ρ has best integrated performance in terms of both energy and delay.

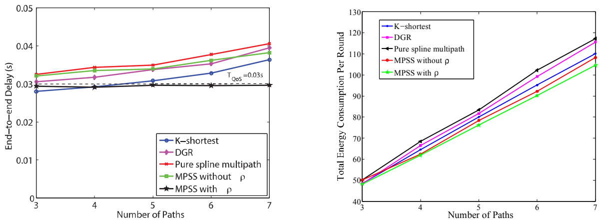


(a) Performance of normalized end-to-end delay

(b) Performance of total energy consumption per round

Fig. 6. Comparison of delay and energy performance varying T_{QoS}

Fig. 7 shows the impact of path number on five schemes in terms of energy and delay. When the number of paths is increased from 3 to 7, both energy and delay are enlarged for all of the schemes. It is because the inter-path interference is increased when more paths are constructed. This situation becomes even worse for K-shortest path scheme, where multiple paths are more densely established than other schemes.



(a) End-to-end delay comparison

(b) Comparison of total energy consumption per round

Fig. 7. Comparison Delay and energy with QoS delay requirement.

E. Realization of MPMS

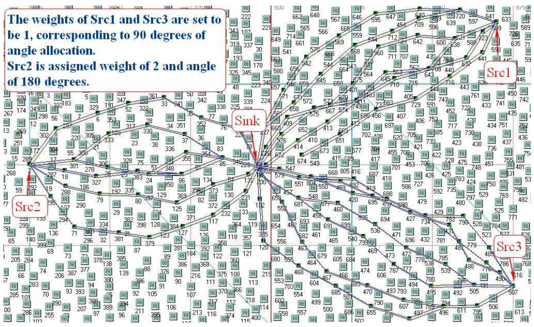


Fig. 8. Three groups of multipaths constructed for three sources, Src1, Src2 and Src3 with different weights

In our experiment, Src2 has the highest "angle assignment" weight among the three source nodes, which represents Src2 has relatively higher data priority and larger amount of data flow than those of Src1 and Src3 . In comparison, we set the same angle assignment weight to Src1 and Src3. Thus, the multipath formation of Src1 and Src3 is symmetrical while Src 2 occupies 1/2 share of full 360 degree "angle resources", as shown in Fig. 8. Due to the network dynamics, some source node(s) will be deactivated or reactivated, which brings changing issue for the reconstruction of multipath formation in MPMS. Fig. 9 shows a typical example where Src 2 stops to generate multimedia data and the occupied angle resources are released for the left source nodes, facilitating Src 1 and Src 3 to obtain better multipath allocations.

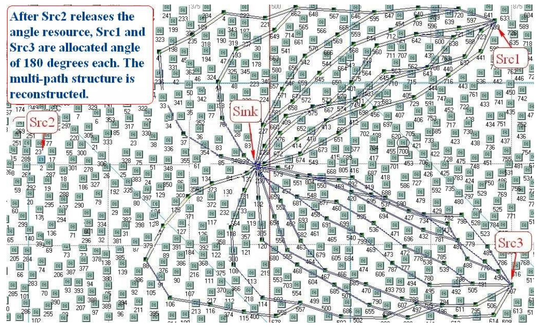


Fig. 9. Illustration of Angle Resource Relocation when one source node quits

V. Conclusion

In this paper, we described the tradeoff of end-to-end delay and energy consumption for WMSNs to support multimedia services with delay QoS requirements. We designed a Multipath Planning for Single-Source based transmissions routing scheme(MPSS) to adjust the transmission range of each hop. We further enhanced our results introducing factor ρ for overcoming the problem induced by the difference between practical next hop and virtual next hop. The simulation results verify that the MPSS can satisfy QoS requirements while providing more energy savings. We also designed a Multipath Planning for Multi-Source routing scheme (MPMS),which allocates angle resources according the angle assignment weight of each source node while dynamically optimizing multipath allocation which some sources are deactivated or reactivated.

Acknowledgment

This work is supported by the National Natural Science Foundation of China (no. 61572220). The work is also supported by the Science and Technology Program of Guangdong Province under Grant no. 2013B091100014.

References

[1] M. Chen, V. C. Leung, S. Mao, and Y. Yuan, "Directional geographical routing for real-time video communications in wireless sensor networks," *Elsevier Journal of Computer Communications*, Vol. 30, No. 17, pp.3368-3383, Nov. 2007.

[2] E. Felemban, C.-G. Lee, and E. Ekic, "Mmspeed: Multipath multi-speed protocol for qos guarantee of reliability and timeliness in wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 5, no. 6, pp. 738–754, Apr 2006.

[3] O. Ghica, G. Trajcevski, P. Scheuermann, N. Valtchanov, and Z. Bischof, "Controlled multi-path routing in sensor networks using bezier curves," *The Computer Journal*, vol. doi:10.1093/comjnl/bxq034, Apr 2010.

[4] L. Galluccio, A. Leonardi, G. Morabito, and S. Palazzo, "A mac/routing cross-layer approach to geographic forwarding in wireless sensor networks," *Elsevier Journal of Ad Hoc Networks*, pp. 872–884, May 2007.

[5] D. Niculescu and B. Nath, "Trajectory based forwarding and its applications," *ACM MobiCom*, pp. 260–272, Jul 2003.

[6] M. Chen, V. C. Leung, S. Mao, Y. Xiao, and I. Chlamtac, "Hybrid geographic routing for flexible energy-delay tradeoff," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 9, pp. 4976–4988, Dec 2009.

[7] G. Feng, T. Korkmaz, "Finding Multi-Constrained Multiple Shortest Paths," *IEEE Transactions on Computers*, Vol. 64, No. 9, pp. 2559-2572, Sep. 2015.

[8] L. Zhou and H. Wang, "Toward blind scheduling in mobile media cloud: fairness, simplicity, and asymptotic optimality," *IEEE Transactions on Multimedia*, Vol. 15,no. 4, pp. 735-746, 2013.

[9] L. Zhou, Z. Yang, J. J. Rodrigues, and M. Guizani, "Exploring blind online scheduling for mobile cloud multimedia services," *IEEE Wireless Communications*, Vol. 20,no. 3, pp. 54-61, 2013.