

Article

Optimizing Retransmission Threshold in Wireless Sensor Networks

Ran Bi ^{1,*}, Yingshu Li ², Guozhen Tan ^{1,†} and Liang Sun ^{1,†}

¹ Department of Computer Science, Dalian University of Technology, Dalian 116000, China; gztan@dlut.edu.cn (G.T.); liangsun@dlut.edu.cn (L.S.)

² Department of Computer Science, Georgia State University, Atlanta, GA 30303, USA; yli@cs.gsu.edu

* Correspondence: biran@dlut.edu.cn; Tel.: +86-411-8470-7544

† Current address: No. 2 Linggong Road, Hi-Tech Zone, Dalian 116000, China.

Academic Editor: Yunchuan Sun

Received: 22 March 2016; Accepted: 29 April 2016; Published: 10 May 2016

Abstract: The retransmission threshold in wireless sensor networks is critical to the latency of data delivery in the networks. However, existing works on data transmission in sensor networks did not consider the optimization of the retransmission threshold, and they simply set the same retransmission threshold for all sensor nodes in advance. The method did not take link quality and delay requirement into account, which decreases the probability of a packet passing its delivery path within a given deadline. This paper investigates the problem of finding optimal retransmission thresholds for relay nodes along a delivery path in a sensor network. The object of optimizing retransmission thresholds is to maximize the summation of the probability of the packet being successfully delivered to the next relay node or destination node in time. A dynamic programming-based distributed algorithm for finding optimal retransmission thresholds for relay nodes along a delivery path in the sensor network is proposed. The time complexity is $O(n\Delta \cdot \max_{1 \leq i \leq n} \{u_i\})$, where u_i is the given upper bound of the retransmission threshold of sensor node i in a given delivery path, n is the length of the delivery path and Δ is the given upper bound of the transmission delay of the delivery path. If Δ is greater than the polynomial, to reduce the time complexity, a linear programming-based $(1 + p_{min})$ -approximation algorithm is proposed. Furthermore, when the ranges of the upper and lower bounds of retransmission thresholds are big enough, a Lagrange multiplier-based distributed $O(1)$ -approximation algorithm with time complexity $O(1)$ is proposed. Experimental results show that the proposed algorithms have better performance.

Keywords: packet delivery; optimal retransmission threshold; in time

1. Introduction

Wireless Sensor Networks (WSNs) have been increasingly deployed for a wide variety of real-time applications, such as industrial Internet-of-Things, emergency response, critical infrastructure monitoring and process measurement and control. In real-time applications, deadline misses in data transmission may bring about irreparable damage [1–4]. For mission-critical tasks, not only the packet delivery deadline should be met, but also the transmission reliability is supposed to be guaranteed. Therefore, providing reliable and timely data delivery in WSNs is crucial to the success of the mission.

In supporting mission critical-tasks, data delivery is required to be timely and reliable, but it is challenging for desirable Quality of Service (QoS). Due to the transmission uncertainties, wireless link dynamics and the queueing dynamics, wireless link qualities in sensor networks can vary at a wide range of timescales [5,6], which also results in node failures and connectivity varying over time [7]. In practical sensor networks, the sensor node may need multiple retransmissions for successfully forwarding a packet at each hop. A maximum number of retransmissions, *i.e.*, retransmission threshold,

is usually set for each node to prevent transmitting a packet over a bad link unlimitedly, and the packet will be eventually dropped by the sender after the maximum number of transmission retries. The success of a packet delivery is probabilistic, and thus, the WSNs are rendered unreliable.

QoS supported by the underlying networks aims to meet end users' satisfaction with the services that the system provides [8]. The QoS provisioning in data delivery can be measured in the form of performance metrics, such as delay, packet loss rate and throughput. There exists a tradeoff among the QoS requirements, such as transmission reliability and deliver delay. Thus, the soft QoS supported can be accomplished, in which the soft QoS refers to achieving the QoS requirements with probability. Many efforts have been made to improve transmission reliability and deliver delay in WSNs, including quality measurement of forwarding data [5,9,10] and real-time routing protocol designing [11–13].

The main methodology applied in the existing works is to exploit a delivery path based on the QoS-aware forwarding quality metrics to improve the performance of packet transmission. However, there exist limited studies on the optimization of the retransmission threshold for each node in a delivery path, which imposes a significant effect on the probability of a packet being successfully transmitted within specified deadline. Most protocols set the same retransmission threshold for all sensor nodes in advance without considering the delay requirement and link quality, so that the delivery performance in terms of latency and reliability is decreased. We use an example inspired by the one in [14] to show the impact of retransmission threshold on the performance, illustrated in Figure 1.

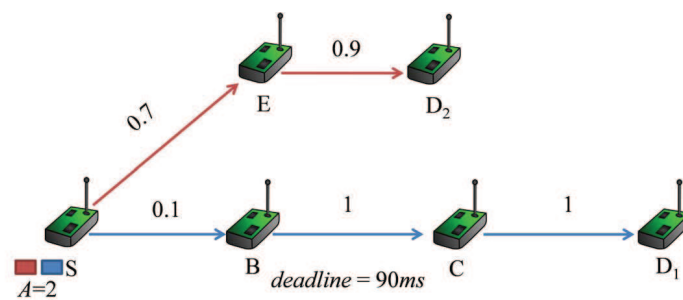


Figure 1. Impact of the retransmission threshold on the probability of packet delivery.

The number associated with each link is the probability for a packet being successfully delivered to the next node through the link, denoted by p_{suc} , which means that on average, $1/p_{suc}$ transmission trials are needed to successfully deliver a packet through the link. The integer variable A denotes the number of packets in the buffer waiting to be served. The delivery path for the first packet is $S \rightarrow B \rightarrow C \rightarrow D_1$, and the path for the second packet is $S \rightarrow E \rightarrow D_2$. We assume that one transmission takes 10 ms, and the deadline of delivering the first packet from node S to node D_1 is 90 ms. For delivering the packet to node D_1 before the deadline, the summation of transmission trials along the path must be no more than nine, that is the result of 90 ms divided by one transmission time of 10 ms. Traditional protocols set the same retransmission threshold for each relay node in advance. If the protocol is set by a high retransmission threshold, such as 29, in the CTP protocol [15], without the consideration of delay requirement, thus the probability for successfully transmitting the first packet through the link $S \rightarrow B$ can be significantly improved. However, this strategy may make the deadlines of both packets expire and consumes more system resource. If the maximum number of transmission trials is three for each node, in view of the summation of transmission trials, there are no more than nine. Then, the probability of the first packet passing the link $S \rightarrow B$ is $1 - 0.9^3 = 0.271$. However, if the retransmission thresholds of the three nodes are set to 6, 0 and 0, that is the maximum numbers of transmission trials are 7, 1 and 1, then the probability is $1 - 0.9^6 = 0.927$. In terms of latency and reliability, there is a tradeoff in determining the retransmission threshold. Therefore, the retransmission threshold should be carefully chosen to achieve desirable QoS provisioning [9].

From the example above, it is obvious that the retransmission threshold has significant impact on the delivery performance in terms of delay and reliability. Few studies focus on the optimization of the retransmission threshold [5,14], and they have the following two major disadvantages. In the feedback control-based framework [5], a heuristic method for adjusting the transmission numbers is proposed, which does not provide the reliability guarantee. In our previous work [14], a greedy-based algorithm for finding optimal retransmission thresholds is proposed, which overlooks the impact of queue length on the forwarding quality. The aforementioned observation motivates us to investigate the problem of finding optimal retransmission thresholds in this paper. The problem is formalized as an integer optimization problem first. Then, a dynamic programming-based distributed algorithm for solving the problem is proposed. It can be run on a sensor node and enables the node to adaptively set the optimal retransmission threshold based on the link quality and the remaining time to deadline. Its time complexity is $O(n\Delta \cdot \max_{1 \leq i \leq n} \{u_i\})$, where u_i is the given upper bound of the retransmission threshold of sensor node i in a given delivery path and n is the length of the delivery path. The complexity depends on delivery delay Δ . If Δ is less than or equal to the polynomial, the complexity of the algorithm is polynomial. Otherwise, the algorithm is not efficient. In this case, a $(1 + p_{min})$ -approximation algorithm is provided based on the linear programming. Furthermore, when the ranges of the upper and lower bounds of the retransmission thresholds are big enough, a Lagrange multiplier-based distributed $O(1)$ -approximation algorithm with time complexity $O(1)$ is developed. The main contributions of the paper are as follows.

- The problem of finding optimal retransmission thresholds for each node along a delivery path is defined and is formalized as an integer optimization problem.
- A Dynamic Programming-based Distributed Algorithm (DPDA) for solving the problem above is proposed; the correctness of the algorithm is proven; and its time and space complexity are analyzed, *i.e.*, $O(n\Delta \cdot \max_{1 \leq i \leq n} \{u_i\})$, and $O(n\Delta \cdot \max_{1 \leq i \leq n} \{u_i\})$.
- In case of the delivery delay Δ being greater than the polynomial, a Linear Programming-based $(1 + p_{min})$ -Approximation Algorithm (LPAA) is proposed.
- Furthermore, in the case of the ranges of the upper and lower bounds of the retransmission thresholds being big enough, a Lagrange Multiplier-based Distributed Approximation Algorithm (LMDAA) with time complexity $O(1)$ is proposed.
- Simulation experiments are conducted to evaluate the proposed algorithms. Simulation results show that the proposed algorithms in the paper have better performance for real-time data delivery.

The rest of this paper is organized as follows. The related works on real-time data delivery are surveyed in Section 2. In Section 3, the problem is described. In Section 4, a dynamic programming-based distributed algorithm for finding optimal retransmission thresholds is provided. Section 5 gives a linear programming-based approximation algorithm, and the ratio bound is analyzed. In Section 6, a Lagrange multiplier-based distributed $O(1)$ -approximation algorithm with time complexity $O(1)$ is proposed. Experimental results are illustrated in Section 7, and Section 8 concludes this paper.

2. Related Works

The real-time performance of packet delivery is a fundamental factor in sensor networks, and many efforts have been made to design efficient real-time routing protocols in WSNs. Most of the existing works addressed the following two challenges. First, it is essential to design the metric, which measures the forwarding quality, such as node forwarding quality [9], link quality [10], data aggregation [2] and energy efficiency [12]. Approximate aggregation for tracking quantiles and range countings is presented in [2], where a dynamic binary tree based deterministic tracking algorithm is proposed. Based on the actual number of data transmissions, link-Quality of Forwarding (QoF) and node-QoF metrics are proposed in [9]. A data-driven-based

link quality prediction approach is provided in [10], which combines the packet reception ratio and the physical-layer information. Considering queuing delay, the Weighted End-to-End Delay (WEED) metric is introduced in [11]; the work focuses on selecting the path with the minimum expected end-to-end delay. In multi-hop wireless networks, the routing decision takes spatial reusability into account to improve the end-to-end throughput [16]. The authors in [17] utilize the expected available bandwidth to capture the logical intra-flow and inter-flow interference, and an isotonic metric of expected delay is proposed. Dependence among links was revealed recently. By exploiting the low correlated forwarding links, link-correlation Opportunistic Routing (OR) is given in [18]. Facing the uncertainties in wireless communication, the multi-timescale adaptation routing protocol is provided in [6], which can adapt to *in situ* delay conditions in routing delay.

To provide real-time packet delivery, QoS provisioning protocols and transmission scheduling protocols have also been extensively studied. The literature [13] studies geographic opportunistic routing protocol for multi-constrained QoS provisioning in WSNs, and the problem is formulated as a multi-objective multi-constraint optimization problem. By differentiating the priorities of user-level applications, the author in [19] develops an optimization tool for balancing the system resources and presents a cluster-based routing protocol. Under the constraints of delay and load balancing requirements, a centralized heuristic algorithm for selecting the path with minimal cost is provided in [20]. To separate the packet of different QoS requirements, a virtual hybrid potential field is introduced in [21]. Considering capacitated multicast routing problem, the work [22] provides the approximation algorithms for multicast k-path routing and multicast k-tree routing. By supporting slot rescheduling, the segmented slot assignment-based method for improving retransmission efficiency is proposed in [23]. Pan *et al.* [24] formulate joint routing and scheduling into an optimization problem and solve the problem with a column generation method. To route around the dead end relay node, the authors in [25] propose the cross-layer protocol, which integrates geographic routing with the contention-based MAC protocol. The author in [26] map the scheduling of data flow to multiprocessor scheduling and prove an upper bound of the end-to-end communication delay.

There exist limited studies on optimizing the retransmission threshold [5,14]. A distributed route maintenance framework is proposed in [5], which enables the link to adjust the retransmission number. However, the proposed heuristic method cannot provide the reliability guarantee theoretically. In our previous work [14], a greedy-based algorithm for finding optimal retransmission thresholds is proposed, which overlooks the impact of queue length on the forwarding quality. To overcome the shortage of the method, we investigate the problem of finding optimal retransmission thresholds in this paper.

3. Problem Description

In this section, the problem of finding optimal retransmission thresholds for each node along a delivery path is defined and is formalized as a general integer optimization problem. We prove that the problem is NP-hard.

3.1. Problem Definition

Suppose that the end-to-end path is $P = j_1, j_2, \dots, j_{n+1}$, where j_1 and j_{n+1} are source and destination nodes and other j_i 's are relay nodes. There exist the following four attributes for each link $j_i \rightarrow j_{i+1}$.

- transmission time t : This means that once transmission costs time, including encoding and wireless communication. Usually each sensor node takes the same time for once transmission.
- transmission failure probability p_i : p_i denotes the transmission failure probability over the link $j_i \rightarrow j_{i+1}$. It means that p_i is the probability of a transmission failure due to either collisions or bad channel quality when node j_i forwards a packet to node j_{i+1} .

- retransmission threshold K_i : This means that the maximum number of retransmission trials is K_i . It is obvious that the maximum number of transmission retries is $K_i + 1$, and thus, the packet will be eventually dropped by the sender after the $K_i + 1$ transmission retries. It can be known that $1 - p_i^{K_i+1}$ is the probability of a packet being successfully delivered to node j_{i+1} .
- retransmission threshold A_i : Each hop delay consists of the transmission delay over the wireless link and the queuing delay in the buffer [9]. A_i denotes the number of the packets queued at node j_i , which means that there exist $A_i + 1$ packets to be forwarded [6].

We define the metric of a given retransmission threshold as the probability of a packet being successfully delivered through the link. Existing works exploit multipath routing to guarantee both reliability and deliver delay in WSNs [13,27,28]. Similarly to [5,9], we aggregate the measure over a path based on the metric and aim at maximizing the summation of the probability of the packet delivered to the next relay node along the given path within the deadline. The path-metric estimates the given path forwarding quality, and it considers both transmission reliability and delay constraint.

The end-to-end delay over a path is the summation of the delays of all of the hops along the path. Let δ denote the delay constraint. Then, the problem of finding the optimal retransmission thresholds for each node can be formulated as the following integer optimization problem, where L_i and U_i are the given lower and upper bounds of the retransmission threshold of node i .

$$\begin{aligned} \max \quad & \sum_{i=1}^n (1 - p_i^{K_i+1}) \\ \text{s.t.} \quad & \sum_{i=1}^n (A_i + 1)(K_i + 1)t \leq \delta \\ & L_i \leq K_i \leq U_i, K_i \in \mathbb{Z}, i \in \{1, 2, \dots, n\} \end{aligned} \quad (1)$$

Each hop delay consists of the transmission delay and the queuing delay in the buffer; thus, the maximum delay latency for each hop is $(A_i + 1)(K_i + 1)t$. The first inequality constraint means that the deliver delay of the last packet served at source node j_1 is no more than the deadline constraint δ . The second inequalities mean that the retransmission thresholds should be bounded in a given interval.

For the convenience of formalization and analysis, let Δ denote δ/t . For any $i \in \{1, 2, \dots, n\}$, let $a_i = A_i + 1, k_i = K_i + 1, l_i = L_i + 1$ and $u_i = U_i + 1$; then, the problem above is equivalent to the following integer optimization problem.

$$\begin{aligned} \min \quad & \sum_{i=1}^n p_i^{k_i} \\ \text{s.t.} \quad & \sum_{i=1}^n a_i k_i \leq \Delta \\ & l_i \leq k_i \leq u_i, k_i \in \mathbb{Z}, i \in \{1, 2, \dots, n\} \end{aligned} \quad (2)$$

3.2. Computational Complexity Analysis

In this subsection, we construct a 0-1 Knapsack problem [29], and we prove that integer optimization Equation (2) is equivalent to the proposed 0-1 Knapsack problem. The computational complexity of the 0-1 Knapsack problem is NP-hard [29]; thus the hardness of the problem for calculating optimal retransmission thresholds is NP-hard.

Let $S_{i,j}$ denote $p_i^{l_i+j} - p_i^{l_i+j-1}$, where $j \in \{1, 2, \dots, u_i - l_i\}$. It can be known that $p_i^{l_i+j} = \sum_{k=1}^j S_{i,k} + p_i^{l_i}$. Thus, we can formulate a 0-1 Knapsack problem as follows.

$$\begin{aligned}
& \min \sum_{i=1}^n \sum_{j=1}^{u_i-l_i} C_{i,j} S_{i,j} + p_i^{l_i} \\
& \text{s.t. } \sum_{i=1}^n \sum_{j=1}^{u_i-l_i} a_i C_{i,j} + \sum_{i=1}^n a_i l_i \leq \Delta \\
& C_{i,j} \in \{0, 1\}, i \in \{1, 2, \dots, n\}, j \in \{1, \dots, u_i - l_i\}
\end{aligned} \tag{3}$$

Lemma 1. For $\forall i \in \{1, \dots, n\}$, the following inequalities hold if $0 < p_i < 1$ and $l_i < u_i$,

$$p_i^{l_i+j} - p_i^{l_i+j-1} < p_i^{l_i+j+1} - p_i^{l_i+j} < 0 \tag{4}$$

where $j \in \{1, \dots, u_i - l_i\}$.

Proof. Let $g_i(x) = p_i^x$. For any $i \in \{1, \dots, n\}$, it is obvious that $g_i(x)$ is a monotone decreasing function; thus, $p_i^{l_i+j+1} - p_i^{l_i+j} < 0$. Obviously, $p_i(p_i^{l_i+j} - p_i^{l_i+j-1}) = (p_i^{l_i+j+1} - p_i^{l_i+j})$. Since $0 < p_i < 1$ and $p_i^{l_i+j+1} - p_i^{l_i+j} < 0$, we have $p_i^{l_i+j} - p_i^{l_i+j-1} < p_i^{l_i+j+1} - p_i^{l_i+j}$. In conclusion, $p_i^{l_i+j} - p_i^{l_i+j-1} < p_i^{l_i+j+1} - p_i^{l_i+j} < 0$. \square

Lemma 2. If $C_{1,1}^*, C_{1,2}^*, \dots, C_{1,u_1-l_1}^*, C_{2,1}^*, \dots, C_{n,u_n-l_n}^*$ are the optimal solutions to Equation (3), then there exist j_1, \dots, j_n , such that:

$$C_{i,k}^* = \begin{cases} 1 & k \leq j_i \\ 0 & k > j_i \end{cases} \tag{5}$$

where $i \in \{1, 2, \dots, n\}, k \in \{1, \dots, u_i - l_i\}, j_i \in \{0, 1, \dots, u_i - l_i\}$.

Proof. The proof is by contradiction. Suppose that $C_{1,1}^*, C_{1,2}^*, \dots, C_{1,u_1-l_1}^*, C_{2,1}^*, \dots, C_{n,u_n-l_n}^*$ are the optimal solutions to Equation (3) and that they enable the objective function value to achieve the minimum. For given m , if there is no integer in $\{0, 1, \dots, u_m - l_m\}$ such that Equation (5) are true, then there must exist integers $r, t, q \in \{1, \dots, u_m - l_m\}$ such that $C_{m,r}^* = 1, C_{m,t}^* = 0, C_{m,q}^* = 1$ and $r < t < q$. We can construct a solution, whose objective function values are less than that of $C_{1,1}^*, \dots, C_{n,u_n-l_n}^*$, by exchanging $C_{m,t}^*$ and $C_{m,q}^*$. Suppose that $C'_{1,1}, C'_{1,2}, \dots, C'_{n,u_n-l_n}$ are derived by merely exchanging $C_{m,t}^*$ and $C_{m,q}^*$.

Firstly, we prove that $C'_{1,1}, C'_{1,2}, \dots, C'_{n,u_n-l_n}$ are feasible solutions of Equation (3). For any $i \in \{1, 2, \dots, n\}$ and $j \in \{1, \dots, u_i - l_i\}$, $C_{i,j}^*$ and $C'_{i,j}$ are 0,1 variables. It is obvious that $\sum_{j=1}^{u_m-l_m} a_m C_{m,j}^* = \sum_{j=1}^{u_m-l_m} a_m C'_{m,j}$. According to the construction of the solution, $C_{i,j}^*$ and $C'_{i,j}$ are identical correspondingly, except for $C_{m,t}^*$ and $C_{m,q}^*$. Since $\sum_{i=1}^n \sum_{j=1}^{u_i-l_i} a_i C_{i,j}^* + \sum_{i=1}^n a_i l_i \leq \Delta$, we have $\sum_{i=1}^n \sum_{j=1}^{u_i-l_i} a_i C'_{i,j} + \sum_{i=1}^n a_i l_i \leq \Delta$. Thus, $C'_{1,1}, C'_{1,2}, \dots, C'_{n,u_n-l_n}$ are feasible solutions.

Let $\varepsilon = \sum_{i=1}^n \sum_{j=1}^{u_i-l_i} C_{i,j}^* S_{i,j} + p_i^{l_i} - \sum_{i=1}^n \sum_{j=1}^{u_i-l_i} C'_{i,j} S_{i,j} + p_i^{l_i}$ be the difference between the values of the two objective functions. It is easily derived that $\varepsilon = \sum_{j=1}^{u_m-l_m} C_{m,j}^* S_{m,j} - \sum_{j=1}^{u_m-l_m} C'_{m,j} S_{m,j} = S_{m,q} - S_{m,t}$. From Lemma 1, we know $\varepsilon > 0$. Then, $C'_{1,1}, \dots, C'_{n,u_n-l_n}$ are more optimal solutions, and it contradicts the fact that $C_{1,1}^*, \dots, C_{n,u_n-l_n}^*$ are the optimal solutions. \square

Theorem 3. Suppose that $C_{1,1}^*, \dots, C_{n,u_n-l_n}^*$ are the optimal solutions to Equation (3). For any $i \in \{1, 2, \dots, n\}$, let $k_i^* = \sum_{j=1}^{u_i-l_i} C_{i,j}^* + l_i$. Then, k_1^*, \dots, k_n^* are the optimal solutions to Equation (2), and the values of the two objective functions are identical.

Proof. Suppose that k'_1, \dots, k'_n are the optimal solutions of Equation (2), then we can construct feasible solutions $C'_{1,1}, \dots, C'_{n,u_n-l_n}$ to Equation (3) as following, where $i \in \{1, 2, \dots, n\}, j \in \{1, \dots, u_i - l_i\}$.

$$C'_{i,j} = \begin{cases} 1 & j \leq k'_i - l_i \\ 0 & j > k'_i - l_i \end{cases} \quad (6)$$

It can be known that $\sum_{i=1}^n \sum_{j=1}^{u_i-l_i} a_i C'_{i,j} + \sum_{i=1}^n a_i l_i = \sum_{i=1}^n \sum_{j=1}^{k'_i-l_i} a_i C'_{i,j} + \sum_{i=1}^n a_i l_i = \sum_{i=1}^n a_i (k'_i - l_i) + \sum_{i=1}^n a_i l_i = \sum_{i=1}^n a_i k'_i$. Since $\sum_{i=1}^n a_i k'_i \leq \Delta$, $C'_{1,1}, \dots, C'_{n,u_n-l_n}$ are the feasible solutions to Equation (3). Based on the definition of $S_{i,j}$, it can be derived that $\sum_{i=1}^n p_i^{l_i+k'_i-l_i} = \sum_{i=1}^n \sum_{j=1}^{k'_i-l_i} S_{i,j} + p_i^{l_i} = \sum_{i=1}^n \sum_{j=1}^{k'_i-l_i} C'_{i,j} S_{i,j} + p_i^{l_i} = \sum_{i=1}^n \sum_{j=1}^{u_i-l_i} C'_{i,j} S_{i,j} + p_i^{l_i}$. Thus, the following inequality can be derived.

$$\min \sum_{i=1}^n p_i^{k_i} = \sum_{i=1}^n p_i^{k'_i} = \sum_{i=1}^n \sum_{j=1}^{u_i-l_i} C'_{i,j} S_{i,j} + p_i^{l_i} \geq \min \sum_{i=1}^n \sum_{j=1}^{u_i-l_i} C_{i,j} S_{i,j} + p_i^{l_i}$$

Therefore, the objective function value of Equation (3) is a lower bound of that of Equation (2).

Since $C^*_{1,1}, \dots, C^*_{n,u_n-l_n}$ are the optimal solutions of Equation (3), there exist j_1, \dots, j_n , such that Equation (5) hold according to Lemma 2. For any i , $\sum_{j=1}^{u_i-l_i} C^*_{i,j} = \sum_{j=1}^{j_i} C^*_{i,j} = j_i$ from Equation (5), then $\sum_{j=1}^{u_i-l_i} C^*_{i,j} S_{i,j} + p_i^{l_i} = \sum_{j=1}^{j_i} S_{i,j} + p_i^{l_i} = p_i^{l_i+j_i}$. Since $k_i^* = \sum_{j=1}^{u_i-l_i} C^*_{i,j} + l_i = j_i + l_i$, we have $\sum_{i=1}^n \sum_{j=1}^{u_i-l_i} a_i C^*_{i,j} + \sum_{i=1}^n a_i l_i = \sum_{i=1}^n \sum_{j=1}^{j_i} a_i C^*_{i,j} + \sum_{i=1}^n a_i l_i = \sum_{i=1}^n a_i j_i + \sum_{i=1}^n a_i l_i = \sum_{i=1}^n a_i (l_i + j_i) = \sum_{i=1}^n a_i k_i^*$. Since $C^*_{1,1}, \dots, C^*_{n,u_n-l_n}$ satisfy all of the constraints of Equation (3), $\sum_{i=1}^n a_i k_i^* \leq \Delta$. For any i , it is obvious that $0 \leq j_i \leq u_i - l_i$. Thus, k_1^*, \dots, k_n^* are the feasible solutions of Equation (2), and hence:

$$\begin{aligned} \min \sum_{i=1}^n \sum_{j=1}^{u_i-l_i} C_{i,j} S_{i,j} + p_i^{l_i} &= \sum_{i=1}^n \sum_{j=1}^{u_i-l_i} C^*_{i,j} S_{i,j} + p_i^{l_i} = \sum_{i=1}^n p_i^{k_i^*} \\ &\geq \min \sum_{i=1}^n p_i^{k_i} \end{aligned} \quad (7)$$

Thus, the objective function value of Equation (2) is a lower bound of that of Equation (3). Based on the analysis above, we can conclude that k_1^*, \dots, k_n^* are the optimal solutions of Equation (2); the values of the two objective functions are identical. \square

Corollary 4. *The hardness of integer optimization Equation (2) is NP-hard. That is, finding optimal retransmission thresholds for each node in a delivery path is NP-hard.*

Proof. Based on Theorem 3, the hardness of Equation (2) is identical to 0-1 programming Equation (3), which is equivalent to the *Knapsack problem*. Since the *Knapsack problem* is NP-hard [29], the hardness of integer optimization Equation (2) is NP-hard. \square

4. Dynamic Programming-Based Distributed Algorithm for Optimal Retransmission Thresholds

As analyzed in Section 3.2, the problem of finding optimal retransmission thresholds is equivalent to the *0-1 Knapsack Problem*. Thus, it can be solved by a dynamic programming-based pseudo-polynomial time algorithm [29]. In this section, we first prove the correctness of the dynamic programming-based algorithm, which can output the optimal resolutions to Equation (3). We design a dynamic programming-based algorithm to solve the proposed *0-1 Knapsack Problem*, and then, according to Theorem 3, the optimal retransmission thresholds can be derived based on the resolutions to Equation (3).

Theorem 5. If $C'_{1,1}, C'_{1,2}, \dots, C'_{n,u_n-l_n}$ are the optimal solutions to Equation (3), then $C'_{1,2}, \dots, C'_{n,u_n-l_n}$ are the optimal solutions to the following problem.

$$\begin{aligned} \min & \sum_{i=2}^n \sum_{j=1}^{u_i-l_i} C_{i,j} S_{i,j} + p_i^{l_i} + \sum_{j=2}^{u_1-l_1} C_{1,j} S_{1,j} + p_1^{l_1} \\ \text{s.t.} & \sum_{i=2}^n \sum_{j=1}^{u_i-l_i} a_i C_{i,j} + \sum_{j=2}^{u_1-l_1} a_1 C_{1,j} + \sum_{i=1}^n a_i l_i \leq \Delta - a_1 C'_{1,1} \\ & C_{i,j} \in \{0, 1\}, j \in \{1, \dots, u_i - l_i\}, i \in \{1, 2, \dots, n\} \end{aligned} \quad (8)$$

Proof. The proof is by contradiction. If $C'_{1,2}, \dots, C'_{n,u_n-l_n}$ are not optimal solutions to Equation (8), suppose that $Z'_{1,2}, \dots, Z'_{n,u_n-l_n}$ are the optimal solutions of Equation (8). Then, $C'_{1,1}, Z'_{1,2}, \dots, Z'_{n,u_n-l_n}$ are better solutions of Equation (3), which leads to a contradiction. The proof demonstrates the correctness of the dynamic programming-based algorithm. \square

To briefly describe the recursive procedure, based on Lemma 2, the following mappings are introduced,

$$\begin{aligned} f : \mathbb{N} \times \mathbb{N} &\rightarrow \mathbb{N}, f(i, j) = \sum_{m=0}^{i-1} (u_m - l_m) + j \\ g : \mathbb{N} &\rightarrow \mathbb{N}, g(h) = \min \left\{ i|h - \sum_{m=0}^i (u_m - l_m) \leq 0, i \geq 1 \right\} \end{aligned}$$

where $u_0 = l_0 = 0$. Let $C_{i,j} = C_{f(i,j)}$, $S_{i,j} = S_{f(i,j)}$ and $sum = \sum_{m=1}^n (u_m - l_m)$, then Equation (3) is equivalent to the following knapsack problem.

$$\begin{aligned} \min & \sum_{h=1}^{sum} C_h S_h + \sum_{i=1}^n p_i^{l_i} \\ \text{s.t.} & \sum_{h=1}^{sum} a_{g(h)} C_h + \sum_{i=1}^n a_i l_i \leq \Delta \\ & C_h \in \{0, 1\}, h \in \{1, 2, \dots, sum\} \end{aligned} \quad (9)$$

When the object set is $\{S_h, S_{h+1}, \dots, S_{sum}\}$ and the capacity is bounded by j , we use $m(h, j)$ to express the minimum cost. Then, we derive the recursive equation of $m(h, j)$.

$$m(h, j) = \begin{cases} m(h+1, j), & 0 \leq j - \sum_{h=1}^n a_h l_h < a_{g(h)} \\ \min \left\{ m(h+1, j), m(h+1, j - a_{g(h)}) + S_h \right\}, & j - \sum_{h=1}^n a_h l_h \geq a_{g(h)} \end{cases}$$

Additionally, the initial conditions are as follows.

$$m(sum, j) = \begin{cases} 0 & 0 \leq j - \sum_{h=1}^n a_h l_h < a_n \\ S_{sum} & j - \sum_{h=1}^n a_h l_h \geq a_n \end{cases}$$

Suppose that node j_i needs to forward a packet with delivery delay Δ' and that the end-to-end path is j_i, j_{i+1}, \dots, j_n . Since the packet is the first one queued at node j_i , $a_i = 1$. Therefore, finding the optimal retransmission threshold k_i for the packet over link from j_i to j_{i+1} is equivalent to solving the following integer optimization problem.

$$\begin{aligned}
& \min \sum_{v=i}^n p_v^{k_v} \\
& \text{s.t.} \quad \sum_{v=i+1}^n a_v k_v + k_i \leq \Delta' \\
& \quad \quad l_v \leq k_v \leq u_v, k_v \in Z, v \in \{i, i+1, \dots, n\}
\end{aligned} \tag{10}$$

According to Theorem 3, the above Equation (10) is equivalent to the following knapsack Equation (11).

$$\begin{aligned}
& \min \quad \sum_{h=u_{i-1}-l_{i-1}+1}^{sum} C_h S_h + \sum_{v=i}^n p_v^{l_v} \\
& \text{s.t.} \quad \sum_{h=u_{i-1}-l_{i-1}+1}^{sum} a_{g(h)} C_h + \sum_{v=i}^n a_v l_v \leq \Delta' \\
& \quad \quad C_h \in \{0, 1\}, h \in \{u_{i-1} - l_{i-1} + 1, \dots, sum\}
\end{aligned} \tag{11}$$

Based on the above analysis, knapsack Equation (11) is a special instance of Equation (9), where the ‘object’ set is $\{S_{u_{i-1}-l_{i-1}+1}, \dots, S_{sum}\}$ and the ‘capacity’ is bounded by Δ' . Suppose $C'_{u_{i-1}-l_{i-1}+1}, \dots, C'_{sum}$ are the optimal solutions of Equation (11), then $\sum_{h=u_{i-1}-l_{i-1}+1}^{u_i-l_i} C'_h$ is the optimal retransmission threshold of node j_i . The message exchanges among network nodes are enhanced to carry necessary information of link quality and queue length [6,11], so that each node can independently calculate the optimal retransmission threshold based on the path forwarding quality and the remaining time to deadline. Such a property enables the node to adaptively set the optimal retransmission threshold in a scalable manner. The proposed dynamic-based distributed algorithm for the optimal retransmission threshold is described in Algorithm 1.

We present the analysis for the computational complexity of Algorithm 1. The recurrence process of the algorithm yields the time computation of $(\Delta' - \sum_{h=i}^n a_h l_h) \cdot (\sum_{h=i}^n u_h - l_i)$. Thus, the time complexity of the proposed distributed algorithm is $O(n\Delta \cdot \max_{1 \leq i \leq n} \{u_i\})$. Space complexity can be similarly analyzed, that is $O(n\Delta \cdot \max_{1 \leq i \leq n} \{u_i\})$.

5. Linear Programming-Based Approximation Algorithm

The dynamic programming-based algorithm proposed in Section 4 is not efficient, if Δ is greater than the polynomial function with respect to n . In this section, we construct a linear programming problem and prove that the solution of the proposed problem can be used to construct the approximate solution of integer optimization Equation (2). Additionally, then, a linear programming-based $(1 + p_{min})$ -approximation algorithm is provided.

5.1. Mathematical Foundations

According to p_i, l_i and u_i , piecewise linear function $f_i(z)$ is defined as follows, where $h \in \{0, 1, \dots, u_i - l_i - 1\}$. We use an example to illustrate the defined function, as depicted in Figure 2.

$$f_i(z) = \begin{cases} (z - l_i - h)(p_i^{l_i+h+1} - p_i^{l_i+h}) + p_i^{l_i+h}, & z \in (l_i + h, l_i + h + 1) \\ p_i^z, & z \in \{l_i, l_i + 1, \dots, u_i\} \end{cases}$$

Lemma 6. For any $i \in \{1, 2, \dots, n\}$, $f_i(z)$ is a convex function.

Algorithm 1: Dynamic Programming-based Distributed Algorithm for the optimal retransmission threshold (DPDA).

Input: $U_{i-1}, L_{i-1}, T = \{T_i, T_{i+1}, \dots, T_n\}$, for $h = i, i + 1, \dots, n; k_i = 0$
 $T_h = \langle p_h, A_h, L_h, U_h \rangle$,
 $Int = \sum_{h=1}^{i-1} u_h - l_h$,
 $sum = \sum_{h=i}^n u_h - l_h$

Output: optimal retransmission threshold of j_i .

for $h = i$ **to** n **do**
| $a_h = A_h + 1, l_h = L_h + 1, u_h = U_h + 1$;
for $h = i$ **to** n **do**
| **for** $k = 1$ **to** $u_h - l_h$ **do**
| | $e = f(h, k); S_e = p_h^{l_h+k} - p_h^{l_h+k-1}$;
 $Low = \sum_{h=i}^n a_h l_h$; **for** $j = 0$ **to** $\min(a_n - 1, \Delta' - Low)$ **do**
| | $m(sum, j) = 0$
for $j = a_n$ **to** $\Delta' - Low$ **do**
| | $m(sum, j) = S_{sum}$
for $h = sum - 1$ **to** $Int + 1$ **do**
| | **for** $j = 0$ **to** $\min(a_{g(h)} - 1, \Delta' - Low)$ **do**
| | | $m(h, j) = m(h + 1, j)$
| | | **for** $j = a_{g(h)}$ **to** $\Delta' - Low$ **do**
| | | | $m(h, j) = \min(m(h + 1, j), m(h + 1, j - a_{g(i)} + S_h))$
if $\Delta' - Low < a_i$ **then**
| | | $m(Int + 1, \Delta' - Low) = m(Int + 2, \Delta' - Low)$;
| | | **else**
| | | | $min = \min(m(Int + 2, \Delta' - Low), m(Int + 2, \Delta' - Low - a_i) + S_{Int+1})$
| | | | $m(Int, \Delta' - Low) = min$;
for $h = Int + 1$ **to** $Int + u_i - l_i$ **do**
| | | **if** $m(h, \Delta' - \sum_{k=Int+1}^{h-1} a_i C_k)$
| | | | $= m(h + 1, \Delta' - \sum_{k=Int+1}^{h-1} a_i C_k)$ **then**
| | | | | $C_{Int+1} = 0$; **else**
| | | | | $C_{Int+1} = 1, k_i = k_i + C_{Int+1}$;
return $k_i - 1$;

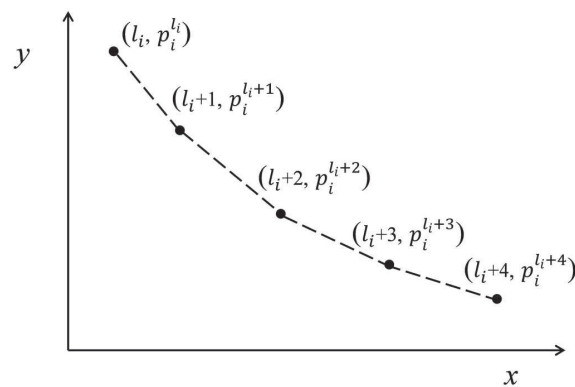


Figure 2. Example of a piecewise linear function.

Proof. For any i , since $f_i(z)$ is a piecewise linear function, $f_i''(z) = 0$, where $z \in (l_i + h, l_i + h + 1)$ and $h \in \{0, 1, \dots, u_i - l_i - 1\}$. Thus, $f_i(z)$ satisfies the sufficient condition of convexity function if $z \in (l_i + h, l_i + h + 1)$. Now, we prove that $f_i(z)$ is a convex function if $z \in \{l_i, l_i + 1, \dots, u_i\}$.

Suppose that $z = \sum_{h=1}^n \lambda_h z_h$, where $z \in \{l_i, l_i + 1, \dots, u_i\}$, $\lambda_1, \dots, \lambda_n \in [0, 1]$, $z_1, \dots, z_n \in [l_i, u_i]$ and $\sum_{h=1}^n \lambda_h = 1$. Since $(p_i^x)'' > 0$, p_i^x is a convex function. Thus, $p_i^z \leq \sum_{h=1}^n \lambda_h p_i^{z_h}$. For any z_h , there must exist $j_h \in \{0, 1, \dots, u_i - l_i - 1\}$, such that $z_h \in [l_i + j_h, l_i + j_h + 1]$.

From the definition of $f_i(z)$, $f_i(z_h) = (z_h - l_i - j_h)(p_i^{l_i+j_h+1} - p_i^{l_i+j_h}) + p_i^{l_i+j_h}$. Let $b'_{i,z_h} = 1 + l_i + j_h - z_h$, $b''_{i,z_h} = z_h - l_i - j_h$, we have $f_i(z_h) = b'_{i,z_h} p_i^{l_i+j_h} + b''_{i,z_h} p_i^{l_i+j_h+1}$, $b'_{i,z_h}, b''_{i,z_h} \in [0, 1]$ and $b'_{i,z_h} + b''_{i,z_h} = 1$. Since $z_h = b'_{i,z_h}(l_i + j_h) + b''_{i,z_h}(l_i + j_h + 1)$, then $p_i^{z_h} \leq f_i(z_h)$ according to the convexity of p_i^x . Based on the above analysis, it can be derived that $p_i^z \leq \sum_{h=1}^n \lambda_h p_i^{z_h} \leq \sum_{h=1}^n \lambda_h f_i(z_h)$.

In conclusion, for any $i \in \{1, 2, \dots, n\}$, $f_i(z)$ is a convex function. \square

According to the constraints of Equation (2), a general optimization problem can be formulated as follows.

$$\begin{aligned} \min & \sum_{i=1}^n f_i(z_i) \\ \text{s.t.} & \sum_{i=1}^n a_i z_i \leq \Delta \\ & l_i \leq z_i \leq u_i, i \in \{1, 2, \dots, n\} \end{aligned} \tag{12}$$

Theorem 7. The objective function value of Equation (12) is a lower bound of that of Equation (2), which implies that a feasible solution can be derived based on the optimal resolutions of Equation (12).

Proof. Suppose that k'_1, k'_2, \dots, k'_n are the optimal solutions to Equation (2). For any $i \in \{1, 2, \dots, n\}$, let $x'_i = k'_i$. Thus, x'_1, x'_2, \dots, x'_n are feasible solutions to Equation (12), and hence:

$$\min \sum_{i=1}^n p_i^{k_i} = \sum_{i=1}^n p_i^{k'_i} = \sum_{i=1}^n f_i(x'_i) \geq \min \sum_{i=1}^n f_i(z_i)$$

Therefore, the objective function value of Equation (12) is a lower bound of that of Equation (2). \square

Theorem 8. The proposed optimization Equation (12) is equivalent to the following linear programming problem, and then, the general optimization Equation (12) can be solved by the linear programming technique.

$$\begin{aligned} \min & \sum_{i=1}^n \sum_{j=0}^{u_i-l_i} \lambda_{i,j} p_i^{l_i+j} \\ \text{s.t.} & \sum_{i=1}^n a_i \sum_{j=0}^{u_i-l_i} \lambda_{i,j} (l_i + j) \leq \Delta \\ & \sum_{j=0}^{u_i-l_i} \lambda_{i,j} = 1, 0 \leq \lambda_{i,j} \leq 1, \\ & j \in \{0, \dots, u_i - l_i\}, i \in \{1, 2, \dots, n\} \end{aligned} \tag{13}$$

Proof. For any $i \in \{1, 2, \dots, n\}$, we have $l_i \leq \sum_{j=0}^{u_i-l_i} \lambda_{i,j} (l_i + j) \leq u_i$. It is easily known that for any $z_i \in [l_i, u_i]$, there exist $\lambda_{i,0}, \dots, \lambda_{i,u_i-l_i} \in [0, 1]$, such that $z_i = \sum_{j=0}^{u_i-l_i} \lambda_{i,j} (l_i + j)$ and $\sum_{j=0}^{u_i-l_i} \lambda_{i,j} = 1$.

Additionally, similarly for any $\lambda_{i,0}, \dots, \lambda_{i,u_i-l_i} \in [0, 1]$, if $\sum_{j=0}^{u_i-l_i} \lambda_{i,j} = 1$, there must exist $z_i \in [l_i, u_i]$, such that $z_i = \sum_{j=0}^{u_i-l_i} \lambda_{i,j}(l_i + j)$.

Suppose that $\lambda_{1,0}, \lambda_{1,1}, \dots, \lambda_{2,0}, \dots, \lambda_{n,u_n-l_n}$ are feasible solutions of Equation (13). For any $i \in \{1, 2, \dots, n\}$, let $z_i = \sum_{j=0}^{u_i-l_i} \lambda_{i,j}(l_i + j)$. Thus, z_1, \dots, z_n are feasible solutions of Equation (12). According to the convexity of $f_i(z)$, we have $f_i(z_i) \leq \sum_{j=0}^{u_i-l_i} \lambda_{i,j}f_i(l_i + j) = \sum_{j=0}^{u_i-l_i} \lambda_{i,j}p_i^{l_i+j}$, and hence:

$$\min \sum_{i=1}^n f_i(z_i) \leq \min \sum_{i=1}^n \sum_{j=0}^{u_i-l_i} \lambda_{i,j}p_i^{l_i+j} \tag{14}$$

Suppose that z'_1, \dots, z'_n are the optimal solutions to Equation (12); there must exist n integers j_1, \dots, j_n , such that $z'_i \in [l_i + j_i, l_i + j_i + 1]$. For any $i \in \{1, 2, \dots, n\}$, let $\lambda'_{i,j_i} = 1 + l_i + j_i - z'_i$, $\lambda'_{i,j_i+1} = z'_i - l_i - j_i$, and the others $\lambda'_{i,j}$ s are zero. Obviously, $\lambda'_{1,0}, \lambda'_{1,1}, \dots, \lambda'_{n,u_n-l_n}$ are feasible solutions of Equation (13). Based on the definition of $f_i(x)$, $f_i(z'_i) = (z'_i - l_i - j_i)(p_i^{l_i+j_i+1} - p_i^{l_i+j_i}) + p_i^{l_i+j_i} = \lambda'_{i,j_i}p_i^{l_i+j_i} + \lambda'_{i,j_i+1}p_i^{l_i+j_i+1} = \sum_{j=0}^{u_i-l_i} \lambda'_{i,j}p_i^{l_i+j}$. Thus, we can have the following formula.

$$\min \sum_{i=1}^n \sum_{j=0}^{u_i-l_i} \lambda_{i,j}p_i^{l_i+j} \leq \sum_{i=1}^n \sum_{j=0}^{u_i-l_i} \lambda'_{i,j}p_i^{l_i+j} = \sum_{i=1}^n f_i(z'_i) = \min \sum_{i=1}^n f_i(z_i) \tag{15}$$

In conclusion, optimization Equation (12) is equivalent to linear programming Equation (13). □

5.2. Linear Programming-Based Approximation Algorithm

Suppose that $\lambda'_{1,0}, \lambda'_{1,1}, \dots, \lambda'_{n,u_n-l_n}$ are the optimal solutions to Equation (13). For any $i \in \{1, 2, \dots, n\}$, let $z'_i = \sum_{j=0}^{u_i-l_i} \lambda'_{i,j}(l_i + j)$. Thus, z'_1, z'_2, \dots, z'_n are the optimal solutions to Equation (12) according to the proof of Theorem 8. To obtain the feasible solutions of integer optimization Equation (2), rounding optimal fractional solutions is a natural idea [29]. The following theorem guarantees that the ratio bound of the rounding approach is $1 + p_{\min}$, where p_{\min} is the minimum of the probabilities of one transmission failure along the given end-to-end path.

Theorem 9. For any $i \in \{1, 2, \dots, n\}$, the ratio bound of the rounding approach is $1 + p_{\min}$, if $\lfloor z'_i \rfloor \geq \frac{\ln p_{\min}}{\ln p_i}$, where $p_{\min} = \min\{p_1, p_2, \dots, p_n\}$.

Proof. Suppose that z'_1, z'_2, \dots, z'_n are the optimal solutions to Equation (12), which are derived from the optimal solutions of linear programming Equation (13). Denote k'_1, k'_2, \dots, k'_n as the optimal solutions to Equation (1). From Theorem 7, the approximation ratio r satisfies the following formula.

$$r = \frac{\sum_{i=1}^n 1 - p_i^{k'_i}}{\sum_{i=1}^n 1 - p_i^{\lfloor z'_i \rfloor}} = \frac{n - \sum_{i=1}^n p_i^{k'_i}}{n - \sum_{i=1}^n p_i^{\lfloor z'_i \rfloor}} \leq \frac{n - \sum_{i=1}^n f_i(z'_i)}{n - \sum_{i=1}^n p_i^{\lfloor z'_i \rfloor}} \tag{16}$$

According to the proof of Theorem 8, for any $i \in \{1, 2, \dots, n\}$, there must exist integer j_i in $\{0, 1, \dots, u_i - l_i - 1\}$, such that $z'_i \in [l_i + j_i, l_i + j_i + 1]$. Based on the convexity of p_i^x , we know that:

$$f_i(z'_i) = (1 + l_i + h - z'_i)p_i^{l_i+h} + (z'_i - l_i - h)p_i^{l_i+h+1} \geq p_i^{z'_i}$$

Let $p_{\min} = \min\{p_1, p_2, \dots, p_n\}$; we have:

$$n - \sum_{i=1}^n f_i(z'_i) \leq n - \sum_{i=1}^n p_i^{z'_i} < n - \sum_{i=1}^n p_i^{\lfloor z'_i \rfloor + 1} \leq n - p_{\min} \sum_{i=1}^n p_i^{\lfloor z'_i \rfloor}$$

Additionally, then, the approximation ratio r satisfies the following formula:

$$\begin{aligned}
 r < \frac{n - p_{\min} \sum_{i=1}^n p_i^{\lfloor z'_i \rfloor}}{n - \sum_{i=1}^n p_i^{\lfloor z'_i \rfloor}} &= \frac{n - np_{\min} + np_{\min} - p_{\min} \sum_{i=1}^n p_i^{\lfloor z'_i \rfloor}}{n - \sum_{i=1}^n p_i^{\lfloor z'_i \rfloor}} \\
 &= \frac{n - np_{\min} + p_{\min} \left(n - \sum_{i=1}^n p_i^{\lfloor z'_i \rfloor} \right)}{n - \sum_{i=1}^n p_i^{\lfloor z'_i \rfloor}} \\
 &= \frac{n - \sum_{i=1}^n p_{\min}}{n - \sum_{i=1}^n p_i^{\lfloor z'_i \rfloor}} + p_{\min}
 \end{aligned} \tag{17}$$

For any $i \in \{1, 2, \dots, n\}$, since $\lfloor z'_i \rfloor \geq \frac{\ln p_{\min}}{\ln p_i}$, $p_i^{\lfloor z'_i \rfloor} \leq p_{\min}$ and $\frac{n - \sum_{i=1}^n p_{\min}}{n - \sum_{i=1}^n p_i^{\lfloor z'_i \rfloor}} < 1$. In conclusion, the ratio bound is $1 + p_{\min}$. \square

The link properties vary as a result of environmental conditions changing at a longer timescale [6,15], then the difference of failure transmission probability among links is little. Thus, the premises of Theorem 9 are always satisfied in practical sensor networks. Since the objective function value of Equation (12) is decreasing with respect to z_i , its optimum value can be achieved if and only if $\sum_{i=1}^n a_i z_i = \Delta$. For a general linear programming problem, finding optimal solutions incurs high computational overhead. However, it can be efficiently solved if the problem is to optimize a linear function subject to linear equality constraints [30].

The linear programming-based distributed approximation algorithm running at each sensor node is described in Algorithm 2, which is derived from the convex programming algorithm in [30]. $f_h^L(x)$ and $f_h^R(x)$ are the left and right derivative of $f_h(x)$, respectively. The proposed algorithm in [30] can yield optimal solutions for the piecewise linear convex function. For any $i \in \{1, 2, \dots, n\}$, the derivative of $f_i(x)$ is discontinuous and constant. According to the proposed algorithm [30], the termination condition can be satisfied when the optimal retransmission threshold is not L_i or U_i . The maximum number for iteration rounds can be assigned in advance. The algorithm can also be stopped if the convergence condition is not achieved. Therefore, figuring out the retransmission threshold of node j_i can be achieved.

6. Lagrange Multiplier-Based Distributed $O(1)$ -Approximation Algorithm

In practical sensor networks, the optimal retransmission thresholds hardly hit the given upper or lower bounds. Therefore, we aim at finding the optimal retransmission thresholds when the ranges of the upper and lower bounds of the retransmission thresholds are big enough. In Section 6.1, we formalize the problem as an integer optimization problem and provide a mathematical method for finding the approximate solution of the problem. Then, a Lagrange multiplier-based distributed $O(1)$ -approximation algorithm with time complexity $O(1)$ is provided.

6.1. Mathematical Foundations

Compared to integer optimization Equation (1), the problem of optimal retransmission thresholds in the case that the ranges of the upper and lower bounds of retransmission thresholds are big enough can be formulated as follows, which implies that the optimization problem has no interval constraints of retransmission thresholds.

Algorithm 2: Linear Programming-Based distributed Approximation Algorithm for the retransmission threshold (LPAA).

Input: $t = \infty, T = \{T_i, T_{i+1}, \dots, T_n\}$,
 $T_h = \langle p_h, A_h, L_h, U_h \rangle$ for $h = i, \dots, n$

Output: retransmission threshold of j_i .

for $h = i$ **to** n **do**
 | $a_h = A_h + 1, l_h = L_h + 1, u_h = U_h + 1$;
 $s_i = f(x_{i1}) - f(x_{i1} + 1), \pi_1 = s_i / a_i$;
for $h = i + 1$ **to** n **do**
 | **if** $h = t$ **then**
 | | $t_h = f(x_{h1}) - s_h x_{h1}$;
 | **else**
 | | $s_h = s_i, t_h = f(x_{h1}) - s_h x_{h1}$;
 $Low = \sum_{h=i}^n a_h l_h$; **for** $h = i + 1$ **to** n **do**
 | **if** $s_h < f_h^R(l_h)$ **then**
 | | $x'_h = l_h$;
 | **else if** $s_h \geq f_h^R(l_h)$ **then**
 | | $x'_h = u_h$;
 | **else**
 | **for** $k = l_h + 1$ **to** $u_h - 1$ **do**
 | | **if** $f_h^L(k) \leq s_h \leq f_h^R(k)$ **then**
 | | | $x'_h = k$;
 | | | **break**;
 | | $y'_h = f_h(x'_h) - s_i x'_h, t_h = f_h(x_{h1}) - s_h x_{h1}$;
 $Min = y'_{i+1} - t_{i+1}$ **for** $h = i + 1$ **to** $n - 1$ **do**
 | | **if** $Min \geq y'_{h+1} - t_{h+1}$ **then**
 | | | $Min = y'_{h+1} - t_{h+1}, temp = h + 1$;
if $Min = 0$ **then**
 | **return** $\lfloor x_{i1} \rfloor$;
else if x_{i1} hits the lower bound l_i or the upper bound u_i **then**
 | **return** $x_{i1} - 1$ and stop;
else

According to the convex programming algorithm [30], adjust x_{i1} and x_t ;

$$\begin{aligned} & \max \sum_{i=1}^n (1 - p_i^{k_i}) \\ & \text{s.t. } \sum_{i=1}^n a_i k_i \leq \Delta \\ & k_i \in \mathbb{Z}^+, i \in \{1, 2, \dots, n\} \end{aligned} \quad (18)$$

The following lemma provides a mathematical method to figure out the optimal solutions of Equation (18) in the real number field; that is for any i , k_i is a real number. Lemma 10 is the foundation for the rounding technique.

Lemma 10. Let $\Lambda = \sum_{i=1}^n (1 - p_i^{x_i}) + \omega (\sum_{i=1}^n a_i x_i - \Delta)$. If there exist x'_1, x'_2, \dots, x'_n , such that for any $i \in \{1, 2, \dots, n\}$, the following equations hold:

$$\frac{\partial \Lambda}{\partial x'_i} = 0$$

then x'_1, \dots, x'_n are the optimal solutions to the following problem.

$$\begin{aligned} & \max \sum_{i=1}^n (1 - p_i^{x_i}) \\ & \text{s.t. } \sum_{i=1}^n a_i x_i \leq \Delta \\ & x_i \in R, i \in \{1, 2, \dots, n\} \end{aligned} \quad (19)$$

Proof. For any $i \in \{1, 2, \dots, n\}$, $p_i^{x_i}$ is a decreasing function. By contradiction, it is easily proven that the objective function achieves the maximum if and only if $\sum_{i=1}^n a_i x_i = \Delta$. Since x_1, x_2, \dots, x_n are real numbers and the only constraint is equality, then we can derive the optimal solutions by the Lagrange multiplier method. We integrate the objective function and the constraint multiplied by ω to the following Lagrange function:

$$\Lambda(x_1, x_2, \dots, x_n, \omega) = \sum_{i=1}^n (1 - p_i^{x_i}) + \omega \left(\sum_{i=1}^n a_i x_i - \Delta \right)$$

Since $\sum_{i=1}^n a_i x_i - \Delta = 0$, $\Lambda(x_1, \dots, x_n, \omega)$ is a function with respect to x_1, \dots, x_n . The gradients on x_1, \dots, x_n and ω are as follows:

$$\begin{cases} \frac{\partial \Lambda}{\partial x_1} = -p_1^{x_1} \ln(p_1) + \omega a_1 \\ \vdots \\ \frac{\partial \Lambda}{\partial x_i} = -p_i^{x_i} \ln(p_i) + \omega a_i \\ \vdots \\ \frac{\partial \Lambda}{\partial x_n} = -p_n^{x_n} \ln(p_n) + \omega a_n \\ \frac{\partial \Lambda}{\partial \omega} = \sum_{i=1}^n a_i x_i - \Delta \end{cases} \quad (20)$$

For any $i \in \{1, 2, \dots, n\}$ and $j \in \{1, 2, \dots, n\}$, we have $\frac{\partial^2 \Lambda}{\partial x_i \partial x_j} = 0$ if $i \neq j$. For any $i \in \{1, 2, \dots, n\}$, it is easily derived that $\frac{\partial^2 \Lambda}{\partial x_i^2} = -p_i^{x_i} \ln^2(p_i)$. The Hessian matrix of $\Lambda(x_1, x_2, \dots, x_n)$ is as follows.

$$\frac{\partial^2 \Lambda}{\partial x_i \partial x_j} = - \begin{pmatrix} p_1^{x_1} \ln^2(p_1) & 0 & \dots & \dots & 0 \\ 0 & \ddots & 0 & \dots & \dots \\ \dots & 0 & p_i^{x_i} \ln^2(p_i) & 0 & \dots \\ \dots & \dots & 0 & \ddots & 0 \\ \dots & \dots & \dots & 0 & p_n^{x_n} \ln^2(p_n) \end{pmatrix}$$

Then, the following equalities can be easily derived.

$$\begin{aligned} & \lambda E - \frac{\partial^2 \Lambda(x_1, x_2, \dots, x_n)}{\partial x_i \partial x_j} \\ & = \begin{pmatrix} \lambda + p_1^{x_1} \ln^2(p_1) & 0 & \dots & \dots & 0 \\ 0 & \ddots & 0 & \dots & \dots \\ \dots & 0 & \lambda + p_i^{x_i} \ln^2(p_i) & 0 & \dots \\ \dots & \dots & 0 & \ddots & 0 \\ \dots & \dots & \dots & 0 & \lambda + p_n^{x_n} \ln^2(p_n) \end{pmatrix} \\ & \left| \lambda E - \frac{\partial^2 \Lambda(x_1, x_2, \dots, x_n)}{\partial x_i \partial x_j} \right| = \prod_{i=1}^n (\lambda + p_i^{x_i} \ln^2(p_i)) \end{aligned}$$

Since all of the eigenvalues of $\frac{\partial^2 \Lambda(x_1, \dots, x_n)}{\partial x_i \partial x_j}$ are negative, the Hessian matrix of $\Lambda(x_1, \dots, x_n)$ is negative definite. In conclusion, if there exist x'_1, \dots, x'_n , such that for any $i \in \{1, 2, \dots, n\}$, $\frac{\partial \Lambda}{\partial x'_i} = 0$, then x'_1, \dots, x'_n are the optimal solutions to Equation (19). \square

Theorem 11. x'_1, \dots, x'_n are the optimal solutions to Equation (19), if for any $i \in \{1, 2, \dots, n\}$, x'_i satisfies Equation (21).

$$x'_i = \frac{\Delta - \sum_{i=1}^n a_i \log_{p_i}^{-a_i(\ln p_i)^{-1}}}{\ln p_i \sum_{i=1}^n a_i (\ln p_i)^{-1}} + \log_{p_i}^{-a_i(\ln p_i)^{-1}} \quad (21)$$

Proof. According to the proof of Lemma 10, if x'_1, \dots, x'_n are the solutions of Equation (20), then they are the optimal solutions to Equation (19). If $\frac{\partial \Lambda}{\partial x'_i} = 0$, we have $p_i^{x'_i} \ln(p_i) = \omega a_i$. For any $i \in \{1, 2, \dots, n\}$, since $a_i > 0, 0 < p_i < 1$ and $\omega < 0$, we have that:

$$x'_i = \log_{p_i}^{\omega a_i (\ln p_i)^{-1}} = \log_{p_i}^{-\omega} + \log_{p_i}^{-a_i (\ln p_i)^{-1}}$$

Since $\frac{\partial \Lambda}{\partial \omega} = \sum_{i=1}^n a_i x'_i - \Delta = 0$, it is easily derived that:

$$\sum_{i=1}^n a_i \left(\log_{p_i}^{-\omega} + \log_{p_i}^{-a_i (\ln p_i)^{-1}} \right) = \Delta$$

By simple calculation, we can get the following equations:

$$\sum_{i=1}^n \frac{a_i \ln(-\omega)}{\ln p_i} + \sum_{i=1}^n a_i \log_{p_i}^{-a_i (\ln p_i)^{-1}} = \Delta$$

$$\ln(-\omega) = \frac{\Delta - \sum_{i=1}^n a_i \log_{p_i}^{-a_i (\ln p_i)^{-1}}}{\sum_{i=1}^n a_i (\ln p_i)^{-1}}$$

In conclusion, x'_1, \dots, x'_n are the optimal solutions to Equation (19), if for any $i \in \{1, 2, \dots, n\}$, x'_i satisfies Equation (21). \square

It is obvious that $\log_{p_i}^{-a_i (\ln p_i)^{-1}} = \frac{\ln a_i + \ln(-(\ln p_i)^{-1})}{\ln p_i}$; the following equation can be derived.

$$x'_i = \frac{\Delta - \sum_{i=1}^n a_i \left(\frac{\ln a_i + \ln(-(\ln p_i)^{-1})}{\ln p_i} \right)}{\ln p_i \sum_{i=1}^n a_i (\ln p_i)^{-1}} + \frac{\ln a_i + \ln(-(\ln p_i)^{-1})}{\ln p_i} \quad (22)$$

6.2. Lagrange Multiplier-Based Distributed $O(1)$ -Approximation Algorithm

The computation of the logarithm is undesired to execute on the sensor node; the log values needed can be stored on the sensor node in advance. For example, three arrays $a[99]$, $b[99]$ and $c[99]$ are forwarded to the sensor node at first, where $a[h] = \ln h$, $b[h] = \ln(0.01h)$, $c[h] = \ln(-(\ln 0.01h)^{-1})$ and $h \in \{1, 2, \dots, 99\}$. Then, the log values needed by Equation (22) are stored in the sensor node. The proposed Lagrange multiplier-based distributed approximation algorithm is described in Algorithm 3. It is easily known that the computation cost is $O(n)$, where n is the number of hops from the source node to the destination node. Since the number of sensor nodes along the given end-to-end delivery path is no more than a few dozen, hence the computation complexity of the algorithm is $O(1)$.

Algorithm 3: Lagrange Multiplier-based Distributed Approximation Algorithm for the retransmission threshold (LMDAA).

Input: $T = \{T_i, T_{i+1}, \dots, T_n\}, T_h = \langle p_h, A_h \rangle,$
for $h \in \{i, i+1, \dots, n\},$
 $\Delta, a[99], b[99], c[99], S_1 = 0, S_2 = 0$
Output: retransmission threshold of node $j_i.$
for $h = i$ **to** n **do**
| $a_h = A_h + 1, p_h = 100p_h;$
for $h = i$ **to** n **do**
| $S_1 = S_1 + a_h \left(\frac{a[a_h] + c[p_h]}{b[p_h]} \right), S_2 = S_2 + \frac{a_h}{b[p_h]};$
 $x_i = \frac{\Delta - S_1}{b[p_i]S_2} + \frac{a[a_i] + c[p_i]}{b[p_i]}.$
return $\lfloor x_i \rfloor - 1;$

Similarly to the linear programming-based approximation algorithm, approximate solutions to integer optimization Equation (18) can be derived by rounding the optimal fractional solutions. The following theorem guarantees that the ratio bound of the rounding approach is $1 + p_{\min}$. Due to $0 < p_{\min} < 1$, the ratio bound of the Lagrange multiplier-based approximation algorithm is two.

Theorem 12. Suppose that x'_1, x'_2, \dots, x'_n are the optimal solutions to Equation (19), then $\lfloor x'_1 \rfloor, \dots, \lfloor x'_n \rfloor$ are the feasible solutions to Equation (18). The ratio bound is $1 + p_{\min}$, if for any $i \in \{1, 2, \dots, n\}, \lfloor x'_i \rfloor \geq \frac{\ln p_{\min}}{\ln p_i}$, where $p_{\min} = \min\{p_1, p_2, \dots, p_n\}$.

Proof. Since x'_1, x'_2, \dots, x'_n are the optimal solutions to Equation (19), $\sum_{i=1}^n a_i x'_i = \Delta$, and hence, $\sum_{i=1}^n a_i \lfloor x'_i \rfloor \leq \Delta$. Thus, $\lfloor x'_1 \rfloor, \dots, \lfloor x'_n \rfloor$ are the feasible solutions to Equation (18). Suppose that k'_1, k'_2, \dots, k'_n are the optimal solutions to Equation (18). Obviously, the objective function value of Equation (19) is an upper bound of that of Equation (18). Thus, the following inequalities can be derived.

$$\sum_{i=1}^n (1 - p_i^{k'_i}) \leq \sum_{i=1}^n (1 - p_i^{x'_i}) < \sum_{i=1}^n (1 - p_i^{\lfloor x'_i \rfloor + 1})$$

Let $p_{\min} = \min\{p_1, p_2, \dots, p_n\}$. Thus, $\sum_{i=1}^n (1 - p_i^{k'_i}) < \sum_{i=1}^n (1 - p_{\min} p_i^{\lfloor x'_i \rfloor})$, and hence, the approximation ratio r satisfies the following equation:

$$r = \frac{\sum_{i=1}^n (1 - p_i^{k'_i})}{\sum_{i=1}^n (1 - p_i^{\lfloor x'_i \rfloor})} < \frac{\sum_{i=1}^n (1 - p_{\min} p_i^{\lfloor x'_i \rfloor})}{\sum_{i=1}^n (1 - p_i^{\lfloor x'_i \rfloor})} = \frac{n - n p_{\min}}{n - \sum_{i=1}^n p_i^{\lfloor x'_i \rfloor}} + p_{\min}$$

In conclusion, the ratio bound is $1 + p_{\min}$, if for any $i \in \{1, 2, \dots, n\}, \lfloor x'_i \rfloor \geq \frac{\ln p_{\min}}{\ln p_i}$. Similarly, we know that the premises of Theorem 12 are easily satisfied in practical sensor networks. \square

7. Experiment Evaluation

The effectiveness and efficiency of the proposed algorithms are evaluated through simulations in this section. Several experiments are conducted to demonstrate the relationships between the performance of real-time data delivery and the input parameter, such as deadline, the number of hops of the given delivery path and the number of packets queued at the relay node. The first two group of simulations are carried out by MATLAB, and the third group is implemented with NS2.

One hundred sensor nodes and the sink are randomly deployed into a region of size $200 \text{ m} \times 200 \text{ m}$ (m for meters), and we assume that the sensors have the same transmission radius. In each simulation, source and destination nodes are randomly selected. Each simulation is repeated

100 times and the simulation result corresponds to the average value over 100 times. To understand the benefits of the proposed algorithms, the comparison with the general method is conducted. The main idea of the general method is that we set the same retransmission threshold for all of the sensor nodes.

The first group of experiments is to investigate the Deadline Success Ratio (DSR) of the proposed algorithms, where DSR is the ratio of the packets delivered to the destination before their deadlines. Figure 3a shows the relationship between the deadlines and DSRs. The proposed algorithms have better performance. For example, when the deadline is 0.22 s, the DSRs of LMDAA and general method are 74% and 61%, respectively. Furthermore, we investigate the impact of the number of hops of a delivery path on the DSR. The remaining time to the deadline and link quality have been considered; thus, our algorithms can achieve higher deadline success ratios as shown in Figure 3b. Figure 3c depicts the relationship between the average number of packets queued at relay nodes and DSRs. From Figure 3c, the DSR becomes worse with the increase of the packets queued on the condition that the deadline is a constant. LMDAA is a distributed algorithm and enables the node to adaptively set the optimal retransmission threshold based on the link quality and the remaining time to the deadline. Therefore, LMDAA has better performance in terms of DSR than that of LPAA.

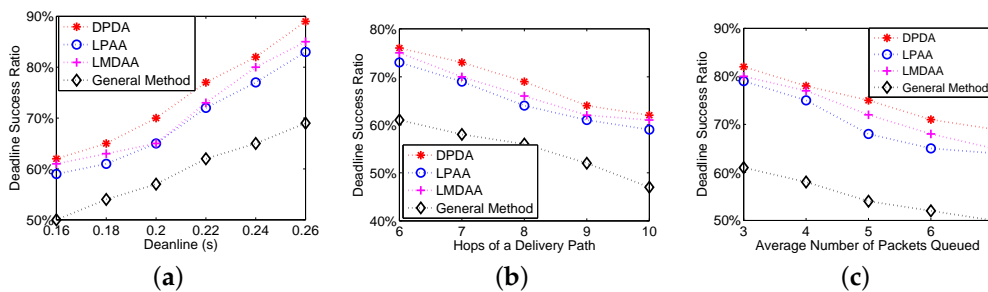


Figure 3. Deadline Success Ratio (DSR) of the proposed algorithms. (a) DSR vs. deadlines; (b) DSR vs. the number of hops; (c) DSR vs. the average number of packets queued.

The second group of experiments is to investigate the Real-Time Ratio (RTR) of the proposed algorithms, where RTR is the ratio of the packets delivered to the destination before their deadlines among the packets successfully delivered to the destination node. Figure 4a shows the relationship between the deadlines and RTRs. As expected, the proposed algorithms can reach higher RTR than that of the general method. Figure 4b depicts the relationship between the number of hops of a delivery path and RTRs. The figure shows that our algorithms have better performance. For example, the real-time ratio of LPAA is more than 60%. Similarly, we investigate the impact of the average number of packets queued at relay nodes on RTR, with results illustrated in Figure 4c. Experimental results show that the proposed algorithms can reduce the deliveries of the packets, which cannot meet their deadlines. Therefore, our algorithms can improve the real-time ratio and energy efficiency.

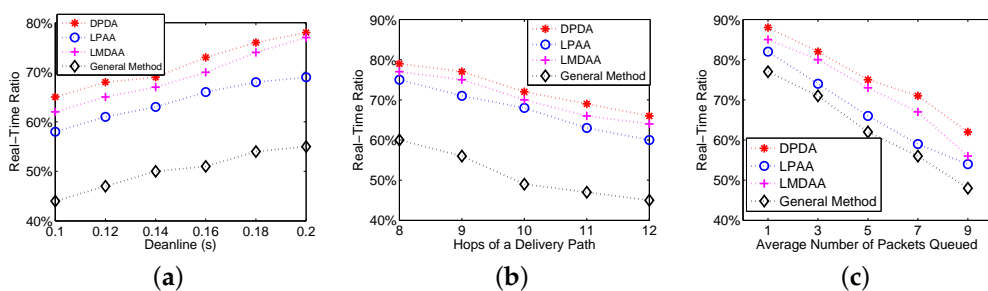


Figure 4. Real-Time Ratio (RTR) of the proposed algorithms. (a) RTR vs. deadlines; (b) RTR vs. number of hops; (c) RTR vs. the average number of packets Queued.

The third group of simulations is implemented with NS2, which is a widely-used simulation tool in wireless sensor networks. From the experimental results of the first two groups of simulations, LMDAA can reveal the effectiveness of the proposed algorithms more neutrally, then we compare the results of LMDAA and the general method. The reasons why DSRs in the simulations implemented by NS2 are lower than those in case of MATLAB are as follows. For identical transmission failure probability, the link quality in NS2 simulation is much worse, which leads to a lower packet delivery ratio. Secondly, we have to set stationary retransmission thresholds in advance in NS2 simulations, and the retransmission thresholds cannot be adaptively set based on the link quality and remaining time during the packet delivery. As shown in Figure 5a–c, the proposed algorithm has better performance in real-time data delivery, in the case of worse wireless links or a severe delivery delay requirement.

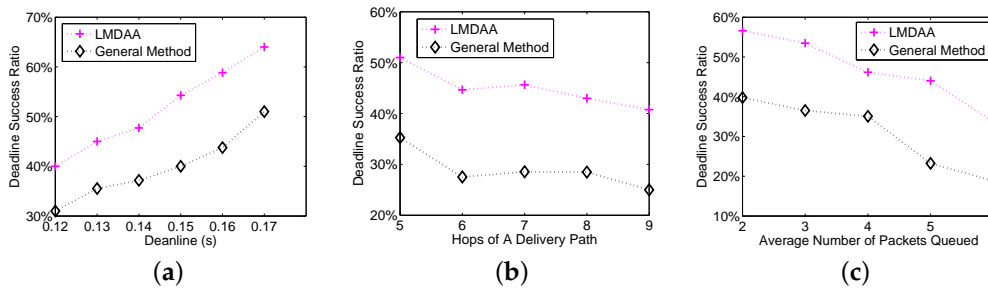


Figure 5. Deadline success ratio by NS2 simulation. (a) DSR vs. deadlines; (b) DSR vs. the number of hops; (c) DSR vs. the average number of packets queued.

The fourth group of experiments is to investigate the computing performance of LPAA and LMDAA, and the correctness of the approximation ratio is verified. We prove that the objective function value of Equation (12) is a lower bound of that of Equation (2), then $\sum_{i=1}^n (1 - f_i(z_i))$ generated by the optimal solution of Equation (12) is an upper bound of Equation (1). In the experiments, the Approximation Ratio (AR) is the ratio of $\sum_{i=1}^n (1 - f_i(z_i))$ generated by the optimal solutions of linear programming Equation (13) to that of the approximated solutions output by LPAA and LMDAA. Figure 6a–c demonstrates the relationships between the approximation ratio and the input parameter. Experimental results show that the objective function value of the approximation results returned by LPAA and LMDAA is very close to that of the optimal ones. Additionally, the proposed approximation algorithms can achieve high accuracy in terms of optimal retransmission thresholds.

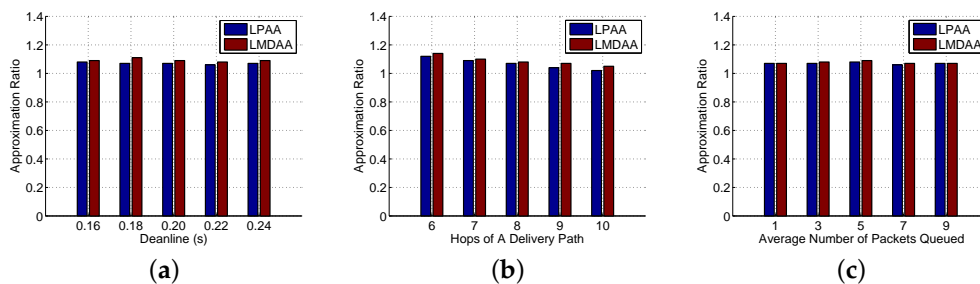


Figure 6. Computing performance of Linear Programming-based $(1 + p_{min})$ -Approximation Algorithm (LPAA) and Lagrange Multiplier-based Distributed Approximation Algorithm (LMDAA). (a) Approximation Ratio (AR) vs. deadlines; (b) AR vs. the number of hops; (c) AR vs. the average number of packets queued.

8. Conclusions

The retransmission threshold in wireless sensor networks is critical to the latency of data transmitting in the networks. The problem of finding optimal retransmission thresholds for each node along a delivery path is defined and is formalized as an integer optimization problem. A dynamic programming-based distributed algorithm for finding the optimal retransmission threshold is proposed. The correctness of the algorithm is proven, and its time and space complexity are analyzed. When the delivery delay Δ is greater than polynomial, a linear programming-based $(1 + p_{min})$ -approximation algorithm is proposed. Furthermore, in the case of the ranges of the upper and lower bounds of the retransmission thresholds being big enough, a Lagrange multiplier-based distributed $O(1)$ -approximation algorithm with time complexity $O(1)$ is proposed. Simulation results show that the proposed algorithms have better performance for real-time data delivery.

Acknowledgments: This work is supported in part by the National Natural Science Foundation of China (61572104, 61402076), the Startup Fund for the Doctoral Program of Liaoning Province (20141023), the Fundamental Research Funds for the Central Universities (DUT15RC(3)088, DUT15QY26) and the open project of the key laboratory of symbolic computation and knowledge engineering in Jilin University (93K172016K11).

Author Contributions: Ran Bi provided the proof of the theorems and designed the algorithms. Yingshu Li supported the theory background and revised the paper. Liang Sun designed the simulations and analyzed the data. Guozhen Tan supported and supervised the research. All of the authors participated in the project, and they read and approved the final manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Cheng, S.; Cai, Z.; Li, J. Curve Query Processing in Wireless Sensor Networks. *IEEE Trans. Veh. Technol.* **2015**, *64*, 5198–5209.
2. He, Z.; Cai, Z.; Cheng, S.; Wang, X. Approximate aggregation for tracking quantiles and range countings in wireless sensor networks. *Theor. Comput. Sci.* **2015**, *607*, 381–390.
3. Cai, Z.; Chen, Z.; Lin, G. A 3.4713-approximation algorithm for the capacitated multicast tree routing problem. *Theor. Comput. Sci.* **2009**, *410*, 5415–5424.
4. Cai, Z.; Goebel, R.; Lin, G. Size-constrained tree partitioning: A story on approximation algorithm design for the multicast k-tree routing problem. In Proceedings of the Third International Conference on Combinatorial Optimization and Applications, Huangshan, China, 10–12 June 2009; pp. 363–374.
5. Lin, S.; Zhou, G.; Al-Hami, M.; Whitehouse, K.; Wu, Y.; Stankovic, J.; He, T.; Wu, X.; Liu, H. Toward stable network performance in wireless sensor networks: A multilevel perspective. *ACM Trans. Sens. Netw.* **2015**, *11*, 42:1–42:26.
6. Liu, X.; Zhang, H.; Xiang, Q.; Che, X.; Ju, X. Taming uncertainties in real-time routing for wireless networked sensing and control. *IEEE Trans. Smart Grid* **2013**, *4*, 288–301.
7. Niu, J.; Cheng, L.; Gu, Y.; Shu, L.; Das, S.K. R3E: Reliable reactive routing enhancement for wireless sensor networks. *IEEE Trans. Ind. Inform.* **2014**, *10*, 784–794.
8. Feng, X. QoS challenges and opportunities in wireless sensor/actuator networks. *Sensors* **2008**, *8*, 1099–1110.
9. Wang, J.; Liu, Y.; He, Y.; Dong, W.; Li, M. QoF: Towards comprehensive path quality measurement in wireless sensor networks. *IEEE Trans. Parallel Distrib. Syst.* **2014**, *25*, 1003–1013.
10. Liu, T.; Cerpa, A. Data-driven link quality prediction using link features. *ACM Trans. Sens. Netw.* **2014**, *10*, doi:10.1145/2530535.
11. Li, H.; Cheng, Y.; Zhou, C.; Zhuang, W. Routing metrics for minimizing end-to-end delay in multiradio multichannel wireless networks. *IEEE Trans. Parallel Distrib. Syst.* **2013**, *24*, 2293–2303.
12. Shi, T.; Cheng, C.; Cai, Z. Adaptive Connected Dominating Set Discovering Algorithm in Energy-Harvest Sensor Networks. In Proceedings of the IEEE International Conference on Computer Communications INFOCOM 2016, San Francisco, CA, USA, 10–15 April 2016.
13. Cheng, L.; Niu, J.; Cao, J.; Das, S.K.; Gu, Y. QoS aware geographic opportunistic routing in wireless sensor networks. *IEEE Trans. Parallel Distrib. Syst.* **2014**, *25*, 1864–1875.

14. Bi, R.; Li, J.; Gao, H.; Chen, Q. Deadline aware retransmission threshold setting protocol in cyber-physical systems. *Int. J. Distrib. Sens. Netw.* **2015**, *2015*, doi:10.1155/2015/271259.
15. Dong, W.; Liu, Y.; He, Y.; Zhu, T. Measurement and analysis on the packet delivery performance in a large scale sensor network. In Proceedings of the IEEE International Conference on Computer Communications INFOCOM 2013, Turin, Italy, 14–19 April 2013; pp. 2679–2687.
16. Meng, T.; Wu, F.; Yang, Z.; Chen, G.; Vasilakos, A.V. Spatial reusability-aware routing in multi-hop wireless networks. *IEEE Trans. Comput.* **2016**, *65*, 244–255.
17. Shi, W.; Shang, S.; Zheng, Y.; Xu, Y. Routing metric of expected delay in multi-radio multi-channel wireless mesh networks. *J. Commun.* **2014**, *9*, 851–858.
18. Wang, S.; Basalamah, A.; Song, M.; Guo, S.; Tobe, Y.; He, T. Link-correlation-aware opportunistic routing in wireless networks. *IEEE Trans. Wirel. Commun.* **2015**, *14*, 47–56.
19. Hammoudeh, M.; Newman, R. Adaptive routing in wireless sensor networks: QoS optimisation for enhanced application performance. *Inf. Fusion* **2015**, *22*, 3–15.
20. Yao, Y.; Cao, Q.; Vasilakos, A. EDAL: An energy-efficient, delay-aware, and lifetime-balancing data collection protocol for heterogeneous wireless sensor networks. *IEEE ACM Trans. Netw.* **2015**, *23*, 810–823.
21. Zhang, J.; Ren, F.; Gao, S.; Yang, H.; Lin, C. Dynamic routing for data integrity and delay differentiated services in wireless sensor networks. *IEEE Trans. Mob. Comput.* **2015**, *14*, 328–343.
22. Cai, Z.; Lin, G.; Xue, G. Improved Approximation Algorithms for the Capacitated Multicast Routing Problem. In Proceedings of 11th Annual International Conference on Computing and Combinatorics COCOON 2005, Kunming, China, 16–29 August 2005; pp. 136–145.
23. Yang, D.; Xu, Y.; Wang, H.; Zheng, T.; Zhang, H.; Zhang, H.; Gidlund, M. Assignment of segmented slots enabling reliable real-time transmission in industrial wireless sensor networks. *IEEE Trans. Ind. Electron.* **2015**, *62*, 3966–3977.
24. Pan, M.; Zhang, C.; Li, P.; Fang, Y. Joint routing and link scheduling for cognitive radio networks under uncertain spectrum supply. In Proceedings of the IEEE International Conference on Computer Communications INFOCOM 2011, Shanghai, China, 10–15 April 2011; pp. 2237–2245.
25. Petrioli, C.; Nati, M.; Casari, P.; Zorzi, M.; Basagni, S. ALBA-R: Load-balancing geographic routing around connectivity holes in wireless sensor networks. *IEEE Trans. Parallel Distrib. Syst.* **2014**, *25*, 529–539.
26. Saifullah, A.; Xu, Y.; Lu, C.; Chen, Y. End-to-end communication delay analysis in industrial wireless networks. *IEEE Trans. Comput.* **2015**, *64*, 1361–1374.
27. Huang, X.; Fang, Y. Multiconstrained QoS multipath routing in wireless sensor networks. *Wirel. Netw.* **2008**, *14*, 465–478.
28. Wang, Z.; Bulut, E.; Szymanski, B.K. Energy efficient collision aware multipath routing for wireless sensor networks. In Proceedings of the IEEE International Conference on Communications, Dresden, Germany, 14–18 June 2009; pp. 1–5.
29. Vazirani, V.V. *Approximation Algorithms*; Springer Science & Business Media: Berlin, Germany, 2013.
30. Dantzig, G.B. *Linear Programming and Extensions*; Princeton University Press: Princeton, NJ, USA, 1998.



© 2016 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC-BY) license (<http://creativecommons.org/licenses/by/4.0/>).