Research Article

Damping Based Traffic Allocation in Wireless Machine-to-Machine Communications Networks

Ruiguo Yu,1,2 Zhenxia Yuan,1,2 Mankun Zhao,1,2 Mei Yu,1,2 and Xu Lu1,2

1 School of Computer Science and Technology, Tianjin University, Tianjin 300072, China
2 Tianjin Key Laboratory of Cognitive Computing and Application, Tianjin University, Tianjin 300072, China

Correspondence should be addressed to Mei Yu; yumei@tju.edu.cn

Received 18 August 2013; Accepted 10 October 2013

Academic Editor: Yan Zhang

Copyright © 2013 Ruiguo Yu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Machine-to-machine (M2M) technology that allows machine devices to communicate directly with each other without human intervention has emerged as a cutting-edge technology for next-generation communication [1]. As its applications become more and more popular, the number of users grows quickly and the quality of service needs to be improved. The traffic allocation problem may lead to many other problems such as low utilization ratio of the bandwidth and the jitter. When we tried to analyze the load on different paths using the simplest multipath routing algorithms, we found that the load changes just like simple harmonic motion, which inspires us to use physic and mathematic methods and tools to analyze the loading balance problems. And then, we designed a new traffic allocation algorithm. Compared with other algorithms, the algorithm proposed in this paper has much wider applicability. And the performance comparison experiment shows that DBTAs can really achieve improvement in decreasing the packet delay variance.

1. Introduction

Machine-to-machine (M2M) technology that allows machine devices to communicate directly with each other without human intervention has emerged as a cutting-edge technology for next-generation communication [1]. M2M technology enables the next generation of wireless sensor networks and global connectivity to billions of processes, devices, and machines through the Internet and it is considered the most promising solution for revolutionizing future applications [2]. However, the wireless M2M networks are composed of a large number of small and power-limited machine devices, which can only transmit low-power signals. Due to the propagation loss between the source devices and the destination devices, it is very challenging to guarantee the quality of service (QoS) requirements of the destination devices [3]. Recently, considerable effort has been put into this problem. Here, we discuss several issues that need to be considered in the design of a framework of wireless M2M communications network to meet the above requirements.

The first issue pertains to the MAC protocol, which plays a key role in determining the network throughput, delay, power consumption, and so forth. A large number of MAC protocols have been proposed for sensor networks and personal area networks, with energy efficiency as the primary design criterion. The most common of these protocols are the contention-based carrier sense multiple access (CSMA) scheme [4] and the schedule-based time division multiple access (TDMA) scheme [5].

The second issue concerns the routing path of data from end devices to an M2M gateway, which is the main content of this paper. Network design has traditionally followed the principle of layering. Previous work in this area has focused largely on the design of link-level protocols [6], and some previous papers address routing for such a network [7]. A large number of studies have shown that the routing protocols based on the criterion of minimum number of hops cannot meet the requirements of M2M network for throughput and guarantee the quality of service.

In recent years, many wireless M2M communications network routing protocols have been proposed: the multiradio link quality source routing protocol [7] is a multichannel routing protocol, which uses weighted cumulative expected time as the routing performance; predictable wireless routing...
protocol [8] by comparing the data packet error rate and other network state parameters to select the optimal path under certain circumstances; single transceiver device and multi-channel routing protocol [9] support multichannel resources in the same area at the same time without disturbing each other so that the network performance can be enhanced; routing protocols combined traditional DSR [10] with the ETX [11] criterion are limited to enhance the throughput, so high-throughput routing protocols make various improvements to enhance the network throughput; based on the ETX criterion RF-aware routing protocol introduces RF-aware routing information as the judgment of the path.

According to how many paths a routing algorithm uses, we can divide all routing algorithms into two classes: single-path routing and multipath routing [12]. Single-path protocol is simple but prone to path overheating (such as multiple clients sending data to the Internet through the same router; all data will be passed along the same path when using a single-path protocol), so that network performance and stability are difficult to be guaranteed; multipath protocols establish multiple paths to transfer data and distribute traffic on multiple paths; thereby they can enhance network throughput, lower packet loss rate transmitted over the network, improve delay and jitter performance indicators, and are better able to achieve network quality of service requirements.

However, most multipath algorithms pay much attention to the multipath discovery and multipath maintenance, while ignoring the traffic allocation problem, which is just the problem we focus on. The traffic allocation problem may lead to many other problems such as low utilization ratio of the bandwidth and the jitter. When we tried to analyze the load on different paths using the simplest multipath routing algorithms we found that the load changes just like simple harmonic motion, which inspires us to use physic and mathematic methods and tools to analyze the loading balance problems [13]. And then we designed a new traffic allocation algorithm. Compared with other algorithms, the algorithms proposed in this paper have much wider applicability.

The structure of this paper is as follows. In Section 1, we first analyze the present algorithms of the wireless M2M communications networks. And an appropriate physical model for the network of multipath traffic allocation problem has been built in Section 2. In Section 3, we combine the conclusions deduced from the physical model with a more mature routing algorithm and design some new multipath routing protocol traffic distribution algorithms for wireless M2M communications networks. The performance comparison experiments are noted in Section 4. And in Section 5 we give the conclusion of this paper.

2. Theoretical Model

2.1. Network Model. We use the graph model to represent wireless M2M communications networks in this paper: wireless network routers and end devices are abstracted as vertices in the graph; the wireless or wired links used by wireless network nodes to communicate with each other are abstracted as edges in the graph. In this model we ignore the phenomenon that the wireless network transmission capacity may be unequal in different directions, because its effects on flow allocation and load balancing are very small; we can simplify the theoretical model and use an undirected graph \( G = \{V, E\} \) to represent the abstraction of wireless M2M communications networks.

There are two original intentions to design new algorithms in this paper. Firstly, reduce the network transmission delay jitter and secondly load balancing. The rules of the action and reaction between the path load condition and its delay are the starting point to design algorithms in this paper. Path load condition is related to the number of data packets per unit time passing, and the number of packets per unit time passing in the path affects the path delay through the impact on the queue length of each node in the path. By the above, we can infer that the queue lengths of the nodes in the path are the bridge linking the load on the path and the path delay.

Therefore, in the theoretical model proposed in this paper, each vertex in the graph uses an integer to describe the queue length of the network node. In addition, the reason for the wait queue of the network node is that a node needs a period of time that cannot be ignored to forward a data packet. Therefore, in the theoretical model, we will also add a constant as the time nodes need to forward a data packet. It needs to be noted that the time for forwarding data packets cannot be ignored, but since the source node can send new packets to the path at any rate, we assume that the time for the source node sending data packets can be ignored. The purpose of this assumption is to design the algorithm in this model that can increase network throughput as much as possible. Its practical significance is that it is usually the wireless network clients that initiated to send data in wireless M2M communications networks; in this case, the first hop node of the path is typically the wireless access point because multiple clients may send data packet to access point simultaneously; we can abstract the clients through the same access point connecting to the wireless M2M communications networks as a source node, and the time of the source node sending data packets can be ignored.

In summary, this theoretical model can be described as

\[
M = \{G, c\}, \\
G = \{V, E\}, \\
V = \{v \mid v = (id, l_{id})\}, \\
E = \{(u, v) \mid u, v \in V\}. 
\]  

The meaning of the model: the model \( M \) is composed of the network topology \( G \) and the constant value \( c \) that represents the time required to forward data packet; network topology graph \( G \) consists of the vertex set \( V \) and the edge set \( E \); the elements in the vertex set \( V \) are a binary group \((id, l_{id})\), where \( id \) is the unique number of vertex and \( l_{id} \) is the current queue length of the network node whose unique number is \( id \); the elements of the set of edges \( E \) are a tuple\((u, v)\); \( u, v \) are different elements in the vertex set \( V \). In the following subsequent sections of this section, we will use this model to calculate the network transmission delay and its changes.
2.2. Theorems

**Theorem 1.** One assumes that none of vertices on the path is in other data packets transmission paths and the queue length is 0 when passing a data packet on the path \( v_1, v_2, \ldots, v_n \); then the forward time of the data packets from \( v_1 \) to \( v_n \) is \( (n - 2)c \).

**Proof.** Based on the assumption that the time for the source node sending data packet to initiate the transmission can be ignored, the residence time of the data packet on the path is the product of the number of intermediate nodes in the path and the time for each intermediate node forwarding the data packet. In order to facilitate the description, \( c \) is identified as the unit of time; that is, \( c = 1 \). When a path is designed for a source node, the lower bound of delay is decided by the number of hops, which is a constant only related to the path itself.

**Theorem 2.** Considering the data packet transmission path \( v_1, v_2, \ldots, v_n \), if the path is used exclusively for \( v_1 \) to \( v_n \), the path delay changes only related to the queue length of \( v_2 \).

**Proof.** Node does not receive data packets from the nodes outside the path, so in unit time, one node can only receive a data packet except \( v_2 \), and at the same time they can forward a packet, so the queue length will remain unchanged and have nothing to do with the path delay variation.

**Theorem 3.** Considering the data packet transmission path \( v_1, v_2, \ldots, v_n \), when the source node \( v_1 \) sends data packets to the target node \( v_n \) with velocity \( v \), the increasing rate of path delay is \( v - 1 \).

**Proof.** Given \( c \) for the unit time, the queue length of node \( v_2 \) changes \( v - 1 \) per unit time; the queuing time variation of the next data packet in the node \( v_2 \) is \( v - 1 \). According to Theorem 2, the path delay also changes \( v - 1 \) per unit time.

2.3. Theoretical Model. According to these three theorems, we can associate network transmission model with vibration model in physics [14, 15]. First of all, we express the path delay as the location \( x \); the rate of change of the path delay must correspond to the speed \( v = dx/dt \); at the same time according to Theorem 3, we can find that the rate of source node sending data packet is proportional to the rate of change of the path delay by making a simple linear transformation. Therefore, the rate of the source node sending data packets to the path can be represented as speed in physics. This paper will describe the following network transmission model by using the physics of vibration model to achieve the design goal below:

(i) the difference of delay time of the two paths \( D_1 - D_2 \) can be described as the oscillator equilibrium position offset \( x \) in the vibration model;

(ii) the rate of change of the difference between the two path delays \( d(D_1 - D_2)/dt \) can be described as the rate of change of the oscillator equilibrium position offset in vibration model \( v = dx/dt \), and according to Theorem 3, we can use the rate of the path receiving new data packet to represent the rate of change.

Therefore, we can make the difference of the two path delays change in accordance with harmonic vibration model in physics by changing the rate of data packets sent to paths in accordance with the following formula:

\[
\frac{d^2 x}{dt^2} + \gamma \frac{dx}{dt} + \omega^2 x = 0.
\]  

(2)

Here, \( k \) is a constant and it will determine the cycle of simple harmonic motion. And \( \frac{d^2 x}{dt^2} \) is the rate of change of the source node sending data packets to different paths in dual-path network transmission model. We can intuitively analyze the meaning of this formula as follows: when the delay on one path is greater than on another path, we will reduce the sending data packet rate of the greater one and increase the rate of the smaller one. This is also the main idea of the most heuristic path selection and flow allocation algorithm.

Obviously, using the formula (2) and other similar traffic shaping policies in the qualitative analysis would lead to endless vibration load on the different paths until the end of the network transmission task. We try to design an algorithm that can make the load vibration gradually flatten and eventually remain stable at the load balancing point. As a motivation, we consider introducing the concept of resistance into the system, so that the load vibration model with its initial state transition similar to the harmonic vibration can transform into damping vibration state in finite time. The physics model used to explain this is shown in Figure 1.

In physics, the significance of the friction is to impede the movement of objects, that is, to hinder the change of objects location. In the network data transfer model, change in the location of objects is related to the change of path delay. Intuitively, the significance of the introduction of friction into the network data model is to maintain the delay on the path unchanged. From the qualitative point of view, the idea to design the algorithm is on the one hand to transfer traffic to the path with lower delay and on the other hand to inhibit this shift to some extent. Through the interaction of these two aspects, we can ultimately achieve the purpose of vibration tending to stationary.

In the following, we will introduce the flow allocation algorithm damping based traffic allocation algorithm (DBTA) in detail. One thing that should be pointed out is that the flow allocation algorithm according to the vibration model designed in this paper is also an algorithm framework that can be used in other network protocols. Therefore, the algorithms proposed in this paper have wider applicability.

3. Damping Based Traffic Allocation Algorithm

3.1. Damping Based Traffic Allocation Algorithm (DBTA). We design a new algorithm to make the network transmission model run in accordance with damping vibration law. Consider the differential equations describing the damping of vibration:

\[
\frac{d^2 x}{dt^2} + \gamma \frac{dx}{dt} + \omega^2 x = 0.
\]  

(3)
In this differential equation, $x$ is the displacement of the oscillator from the equilibrium position, that is, the difference of the two path delays in the network transmission model; $dx/dt$ is the velocity of oscillator, that is, the difference of the source node sending data packet rate to the two paths, while $d^2x/dt^2$ represents the acceleration of the oscillator, that is, the rate of change of the difference of the source node sending data packet rate to the two paths. The core of the proposed algorithm is computing the value, of $d^2x/dt^2$, adjusting the rate sending data packets to two different paths in accordance with the value, and achieving the goal of the adjustment path delay. And $\gamma$ and $\omega_0^2$ are system parameters; that is, damping vibration attenuation rate cycle can be controlled by setting these two parameters, where $\gamma$ is a negative number.

Based on the assumption of the network environment in this paper, there is only one data sender and only one recipient in the network and there are two paths from the sender to the recipient, and then you can get the above three theorems. Because the issue of this paper is the network load oscillation and the purpose of the algorithm is to eliminate this phenomenon, we omitted other complex situations to simplify the problem.

According to Theorem 2, we learn that the path delay is only related to the queue length of the next hop node of the source node in this hypothetical network environment. In dynamic source routing protocol [10], each node only interacts with its next node to receive confirmation, so the design algorithm based on dynamic source routing protocol is easy to access the queue length information of the next hop node of the source node and then to estimate path delay. We have illustrated how to obtain all things needed to be calculated including the path delay and the data packet rates of the source node sending to different paths. Taking into account in practical applications, the source node sending rate usually has an upper limit; it may be decided by the amount of data that needed to be sent or by the hardware capacity. After joining this assumption, the process of the allocation algorithm based on the flow of damping vibration process is as Algorithm 1.

Algorithm 1 is a damping based traffic allocation algorithm. This algorithm makes the load on the path vibrate in the form of damping vibration by adjusting the change rate of the data packets sending to different paths, and ultimately the delay on the two paths is converging. In physics, acceleration is proportional to the size of force and the work in this paper is precisely to control acceleration.

What should be noted is the set of parameters $\omega_0$ and $\gamma$; they can be set according to the actual requirements of the application, given the damped vibration function:

$$x = e^{-\gamma t/2} \left( Ae^{i\omega_0 t} + A e^{-i\omega_0 t} \right).$$

Here, $\omega_0 = \sqrt{\omega_0^2 - \gamma^2}/4$. It can be seen that $\omega_0$ determines the period of vibration. The increases of $\omega_0$ will lead to the shortening of the period of vibration, resulting in severe vibration of the path load in the initial stages of the data transmission. So according to actual needs, if the period of the vibration you can tolerate is $t$ seconds, it will be appropriate to set $\omega_0$ to $2\pi/t$. And the decay rate of amplitude is determined by $\gamma$; the greater the value of $\gamma$ is, the faster the decay is. However, large $\gamma$ will still lead to larger delay difference of the two paths in the vibration stationary position. If the delay difference of the two paths need to be reduced to $x$ after $t$ seconds, you should make $\gamma = -(2 \ln x)/t$.

3.2. Improved Damping Based Traffic Allocation Algorithm (Improved-DBTA). As the actual situations which DBTA applied to are relatively simple, we proposed more complete improved-DBTA on the basis of DBTA to make the algorithm able to achieve better results in a complex network environment. Firstly, improved-DBTA no longer applies only to two paths but to any multiple paths; secondly, improved-DBTA will no longer be simple to use the queue length of the second node on the path as the evaluation of the path but take the path delay which the target node feedback as a basis for adjustment of the rate of sending data to the path.

The delay in the network transmission mainly consists of four parts: the processing delay, queueing delay, sending delay, and propagation delay. To simplify the problem, assuming that the size of packet in the network is similar and the path complexity is analogous to each other, then the processing delay, sending delay, and propagation delay can be regarded as constant. Therefore, the delay of packets in the network is $x = C + x_q$, where $C$ is a constant and $x_q$ is the queueing delay. Therefore, the rate of change of the total propagation delay is decided by the rate of change of queueing delay:

$$\frac{dx}{dt} = \frac{dx_q}{dt}.$$  (5)

Considering a simple path connecting the source node and destination node, the rate of the source node sending data to the path is $r_{\text{input}}$ packages per second, and the maximum output rate of this path is $R$ data packages per second; then there will be $r_{\text{input}} - R$ packets blocked in the queue in a second, so that the path delay of the packets increases $k(r_{\text{input}} - R)$, where $k$ is a constant. Therefore, we can obtain the following relationship:

$$\frac{d^2x}{dt^2} = \frac{d^2x_q}{dt^2} = k \frac{dr_{\text{input}}}{dt}.$$  (6)
According to Newton’s second law, the vector force added to the objects is the reason for the change of momentum of objects:

\[ F = \frac{dp}{dt} = m \frac{dv}{dt} = m \frac{d^2x}{dt^2}. \]  

(7)

If only we considered the situation in one dimension, we can use the sign to indicate the direction of the vector and use the absolute value to indicate the size of the vector; then we can get

\[ F = m \frac{d^2x}{dt^2}. \]  

(8)

Inspired by this, consider the rate of change of the path input rate \( d \ell_{\text{input}} / dt \) as the force \( F \) in physics. Whether it is in theoretical physics or engineering, we can use such a mathematical model to represent damping: the size of the force and speed changes synchronously but in the opposite direction. In mechanics, friction is one such force. In a variety of circumstances, the friction is seen as proportional to the speed of the movement of objects; that is,

\[ F = -cv. \]  

(9)

There, \( c \) is the coefficient of friction. And then consider the situation in the network; assuming that there is a simple path between the source node and destination node, the rate of the source node sending data to this path is \( r_{\text{input}} \) and the maximum output rate of the path is \( R \). If we can find the delay increases on this path in the process of transferring data, then it can be concluded that \( r_{\text{input}} > R \); in this case, the source node should be appropriate to reduce the rate sent to the path; otherwise the delay will be more and more large. Conversely, if the delay on the path decreases, we can know that \( r_{\text{input}} < R \); this time, the source node can appropriately increase the rate sent to this path to improve transmission efficiency. Further, when the path delay variation is greater, the change of sending rate will be greater, but the change should be in the opposite direction. Therefore, the change rate of the velocity of the source node sending data to the path and the rate of change of the path delay have similar links to the movement of objects suffering friction and the object velocity. This also confirms from another point that the network transmission model is linked to the variables in physics.

The process of the improved damping based traffic allocation algorithm in this section is as follows: first, set the equilibrium position of the path delay \( d_{\text{balance}} \) according to the maximum output rate \( R \), then calculate the average path propagation delay \( d \) of the previous period and \( d' \) of the previous two periods, and finally, calculate the increment of the output rate of the source node the next unit of time relative to the last period.

Consider

\[ r_{\text{input}} - r'_{\text{input}} = -\gamma (d - d') - \omega_0^2 (d - d_{\text{balance}}). \]  

(10)

There, \( \gamma \in (0,1) \), \( \omega_0 \in ([\gamma/2,1]) \) are the parameters used to control the damping vibration frequency and decay rate and \( r_{\text{input}} \) is the output rate of the source node in previous period. So far, the network transport protocol, improved-DBTA, used in single-path has finished presentation. In the design of the algorithm, for the convenience of description, we assume that the node outside the path will not send packets to the path and the case is too simple and not realistic. However, the node outside the path sends packets to the path nodes, which works as an external force to the damping vibration of the oscillator, and that may change the vibration amplitude, frequency, and the equilibrium position but still maintains the process of damped vibration. Then, consider extending the algorithm so that it can use multiple paths to transmit packets. Assuming in the path discovery phase, we found \( k \) paths from the source node to destination node: \( p_1, p_2, \ldots, p_k \) and then we represented the delay on the \( k \) paths with the symbol \( d_1, d_2, \ldots, d_k \) and the input rate of the source node to the \( k \) paths with the symbol \( r_1, r_2, \ldots, r_k \), and the average input rate of the \( k \) paths is \( r_{\text{avg}} \) so

\[ r_{\text{avg}} = \frac{1}{k} \sum_{i=1}^{k} r_i. \]  

(11)
Meanwhile we use $d_{\text{avg}}$ to denote the average delay on these $k$ paths; that is

$$d_{\text{avg}} = \frac{1}{k} \sum_{i=1}^{k} d_i.$$  \hspace{1cm} (12)

In addition, the symbol $d_{\text{max}}$ stands for the maximum value of difference between the path delays; that is,

$$d_{\text{max}} = \max_{1 \leq i < j \leq k} |d_i - d_j|.$$  \hspace{1cm} (13)

The goal of the proposed algorithm is to minimize $\Delta d_{\text{avg}}$ and $d_{\text{max}}$. To achieve these two objectives, the algorithm is divided into two parts in this paper. In this paper, these two parts were referred to as flow control section and flow distribution section.

In the flow control section, we take all the $k$ paths as one path and then use the single-path algorithm previously described, where $\Delta d_{\text{avg}}$ is the displacement of the oscillator, and we can control the sending rate of the source node to minimize $\Delta d_{\text{avg}}$.

In the flow distribution section, divide the path into two parts recursively, and then take the difference of the average delay time between these two parts as a displacement of the oscillator to minimize $d_{\text{max}}$. Detailed description of the algorithm is shown in Algorithm 2.

Algorithm 2 is the process of the improved-DBTA. In order to simplify the algorithm, we take the first path as the first part and the rest of the path as the second part during dividing the paths into two parts recursively. If the number of the rest paths is greater than 1, then in the second division, we take the second path as one part and the other paths are composed of the remainder and so on. In the case of using this classification method, we take the difference of the average delay time between the two parts as the displacement relative to the equilibrium position of the damped oscillator, so we can deduce all formulas in the algorithm.

3.3. Adaptive Damping Based Traffic Allocation Algorithm (Adaptive-DBTA). In improved-DBTA, we need to estimate the maximum output rate of the path in advance. It is relatively easy to determine the maximum output rate of the path when the type of nodes is single in the network and the communication between nodes is similar. However, usually the network conditions would be more complex in the actual applications, and, therefore, the maximum output rate of the path is difficult to estimate in advance. Even if the effects of the incorrect estimated maximum output rate of the path on the algorithm are not significant (from the model perspective, the difference between the estimated value and actual value can be filled with a hypothetical data stream, which is equivalent to a constant force with unchanged size and direction applied to the oscillator in the vibration model), we still hope to be able to design a simple algorithm to find the balance point of the sending rate adaptively.

Supposing a path $p$ from the source node to destination node, the source node sends data to the destination node at the rate of $r_1$ and the change of the rate of the path $ps$ delay is $\Delta d_1$ in the unit time. While the source node sends data at the rate of $r_2$ and the change of the rate of the path $ps$ delay is $\Delta d_2$ in the unit time, we can know that if $r_1$ is more away from the maximum output rate $R$ of the path $p$ than $r_2$, then $|\Delta d_1| > |\Delta d_2|$. As described in the previous section, we can assume the difference of the sending rate of source node and the actual maximum output rate of the path directly proportional to the rate of change of the path delay; that is,

$$R - r_1 \times (\Delta d_2) = (R - r_2) \times (\Delta d_1).$$  \hspace{1cm} (14)

It can be introduced by

$$R = \frac{r_1 \times \Delta d_2 - r_2 \times \Delta d_1}{\Delta d_2 - \Delta d_1}. $$  \hspace{1cm} (15)

In adaptive-DBTA, we record the average sending rate of the source node and multipath average delay in the first two periods of time and calculate the equilibrium position in accordance with the above formula, and then, the next period of time, the source node sends data at the sending rate of the calculated equilibrium position. In order to facilitate description, use the symbol $d_i^k$ to express the average delay $d_i$ in the $ith$ period of time on the $kth$ path and use $r_i^k$ to denote the average rate of the source node sending to the $kth$ path. The specific process of algorithm is shown in Algorithm 3.

Algorithm 3 is the process of adaptive-DBTA. The method distributing the load to each path after calculating the total sending rate of previous period is the same as improved-DBTA. In addition, both in improved-DBTA and adaptive-DBTA, the delay information of the paths used in the calculation has some degree of lag, but due to calculation methods, adaptive-DBTA will have a greater lag than improved-DBTA. This is also a major drawback of the adaptive-DBTA. However, adaptive-DBTA does not need to set complex parameters and thus can be more convenient in practical network applications.

4. Experiment

In this paper, we use the NS-2 network simulation software to do the experiments. We do a slight modification on DSR protocol: record the multiple paths in the path finding stage and then use this design algorithm to choose the right path for data packages to be sent. The aim of the simulation experiments is to view whether the flow allocation algorithms designed in this paper are able to reduce the path delay and throughput fluctuations or not.

In the experiments, we established 100 nodes randomly at first and then established 40 streams with different data sending rates. The parameters of NS-2 simulation are as follows: experiment scene: 2000 m $\times$ 2000 m, number of nodes: 100, MAC type: 802.11, IFq type: drop – tail/pri – queue, IFq length: 20, and simulation time: 200 seconds.

In the DBTAs network simulation experiments, we selected some particular paths for comparison to observe the difference between DSR protocol and the flow allocation algorithm in delay when collecting the experimental results.

Figure 2 is the delay comparison chart over time of a path whose absolute value of delay is approximately equal to avoid...
(1) Set the parameters $d_{\text{balance}}$, $\gamma$ and $\omega$, according to the actual application requirements as well as the average maximum output rate of every path;
(2) Set $r_{\text{avg}}_1 = r_2 = \ldots = r_k = 0$, $r'_{\text{avg}}_1 = r'_2 = \ldots = r'_k = 0$;
(3) Set $d'_1 = d'_2 = \ldots = d'_k = 0$, $d''_1 = d''_2 = \ldots = d''_k = 0$;
(4) while There are still data packets need to send do
(5) Calculate $d_{\text{avg}} = \frac{1}{k} \sum_{i=1}^{k} d'_i$, $d''_{\text{avg}} = \frac{1}{k} \sum_{i=1}^{k} d''_i$, and $s_i = \frac{1}{k - i + 1} \sum_{j=i}^{k} d'_j$ in the $t$th period;
(6) Calculate the average sending rate of the source node in the $t$th period: $r_{\text{avg}} = r'_{\text{avg}} - \gamma (d'_{\text{avg}} - d''_{\text{avg}}) - \omega (d''_{\text{avg}} - d_{\text{balance}})$, and calculate the total sending rate of the source node in the $t$th period: $r_{\text{total}} = k \times r_{\text{avg}}$;
(7) Calculate the input rate of the first path: $r_1 = r'_1 + \frac{1}{k} r_{\text{total}} - r'_{\text{total}} - (k - 1) \gamma ((d'_1 - s_1) - (d''_1 - s'_1)) - (k - 1) \omega (d''_1 - s_1)$
for $i = 1, 2, \ldots, k - 1$ and $r_k = r_{\text{total}} - \sum_{j=1}^{k-1} r_j$;
(9) In the $t$th period, the source node sends packet to the $i$th path at the rate of $r_i$. After this period set $r'_i = r_i$, $r''_{\text{avg}} = r_{\text{avg}}$, $d'_i = d_i$, and calculate the average delay $d_i$ of the $i$th path in this period;
(10) Set $t = t + 1$;
(11) end while

Algorithm 2: Improved damping based traffic allocation algorithm (improved-DBTA).

(1) Set the parameters $\gamma$ and $\omega$, according to the actual application requirements;
(2) Gradual increase the rates to each path in the time period of 1 to 3 to send data, and record the sending rate and the average delay of three time periods;
(3) Set $d'_1 = d'_2 = \ldots = d'_k = 0$, $d''_1 = d''_2 = \ldots = d''_k = 0$;
(4) while There are still data packets need to send do
(5) Calculate $d_{\text{avg}} = \frac{1}{k} \sum_{i=1}^{k} d'_i$, $d''_{\text{avg}} = \frac{1}{k} \sum_{i=1}^{k} d''_i$, and $s_i = \frac{1}{k - i + 1} \sum_{j=i}^{k} d'_j$ in the $t$th period;
(6) Calculate the average sending rate of the source node in the $t$th period: $r_{\text{avg}} = \frac{(r''_{\text{avg}} - (d''_{\text{avg}} - d''_{\text{avg}}) - \omega (d''_{\text{avg}} - d_{\text{balance}}))}{(2d''_{\text{avg}} - d''_{\text{avg}} - d''_{\text{avg}})}$, and if the denominator in the equation is 0, then $r_{\text{avg}} = r''_{\text{avg}}$ and calculate the total sending rate of the source node in the $t$th period $r_{\text{total}} = k \times r_{\text{avg}}$;
(7) Calculate the input rate of the first path: $r_1 = r'_1 + \frac{1}{k} r_{\text{total}} - r'_{\text{total}} - (k - 1) \gamma ((d'_1 - s_1) - (d''_1 - s'_1)) - (k - 1) \omega (d''_1 - s_1)$
for $i = 1, 2, \ldots, k - 1$ and $r_k = r_{\text{total}} - \sum_{j=1}^{k-1} r_j$;
(9) In the $t$th period, the source node sends packet to the $i$th path at the rate of $r_i$. After this period set $r'_i = r_i$, $r_{\text{avg}} = r_{\text{avg}}$, $d'_i = d_i$, and calculate the average delay $d_i$ of the $i$th path in this period;
(10) Set $t = t + 1$;
(11) end while

Algorithm 3: Adaptive damping based traffic allocation algorithm (Adaptive-DBTA).

the effects of the absolute value of delay on the volatility. From the experimental results, we can see that the load variation using the DBTA algorithm is much smoother than that when just using DSR protocol. The effect of DBTA algorithm on delay jitter is obvious.

Figure 3 is the throughput comparison chart over time of a path whose throughput is approximately equal to avoid the effects of the size of throughput on the volatility.

In contrast to the delay fluctuations, we still use similar methods and choose the path whose absolute value of delay is approximately equal to avoid the effects of the absolute value of delay on the volatility. The transfer delay result is shown as Figure 4.

In the third part of the experiment, we used our own program written in C language to compare the difference of improved-DBTA and adaptive-DBTA in the delay and
throughput. Similar to the principle of NS-2 principle, we used priority queue to implement event-driven simulation. The aim of the simulation is to study the difference between adaptive-DBTA and improved-DBTA, and the throughput comparison results are shown in Figure 5.

From the above experimental results, we can see, compared with the improved-DBTA, that the throughput jitter of adaptive-DBTA is slightly serious; this result is because each time adaptive-DBTA directly adjusts the sending rate to the equilibrium position calculated, while improved-DBTA gradually adjusts the sending rate to simulate the process of damped vibration. The comparison of these two algorithms in terms of path delay results is shown in Figure 6.

From the above experimental results, we can see, compared with the improved-DBTA algorithm, that the delay jitter of the adaptive-DBTA is slightly serious. And it is related to the experimental results of adaptive-DBTA throughput jitter. The delay of the two algorithms trends to increase, which is also related to the growth trend of throughput.

In addition, we also compared improved-DBTA with DCJOTA in [16], and the purpose of the experiment is to check the difference of effect between the algorithm in this paper and the other algorithms. The results of these two algorithms contrast experiment are shown in Figure 7.

In addition, in the last part of the experiment, we did an experiment to compare the difference in the path delay jitter between improved-DBTA, the DCJOTA, the OPNET in [17], and the COTA in [18]. The path delay comparison results of all these algorithms are shown in Figure 8.

From the above experimental results, we can see that the control effect of the improved-DBTA on the throughput and delay jitter is similar to DCJOTA and COTA algorithms, all of which are indeed achieving the aim to control the delay jitter and are better than the effect of OPNET algorithm.

5. Conclusion

Wireless M2M communications network is a promising network access technology and has broad prospect of applications. The multipath transmission and load balancing problems that related to service quality and throughput of wireless M2M communications network have become a hot spot issue for many researchers. In many studies, people used complex mathematical models to analyze the problem. But we think that the physical model may be more applicative and easy to understand.

In this paper, we analyzed the traffic allocation problem of wireless M2M communications networks and built an appropriate model using physics method. We proposed new traffic allocation algorithms based on damping model, and the algorithms can reduce the network transmission delay jitter and load balancing. Here, we simply applied the damping model of classical mechanics; the more in-depth
Figure 4: The transfer delay of DSR and improved-DBTA.

Figure 5: The throughput of improved-DBTA and adaptive-DBTA.

Figure 6: The transfer delay of improved-DBTA and adaptive-DBTA.

Figure 7: The throughput of improved-DBTA and DCJOTA.
and effective application of the model to analyze and solve problems remains to be explored.

Acknowledgments

This research is supported by the National Science Funds of China Grant no. 61170305 and is part of Project of Education Information Strategy supported by Educational Commission of Tianjin, China, 2013, under Grant no. x2013-005.

References

[1] D. Niyato, L. Xiao, and P. Wang, “Machine-to-machine communications for home energy management system in smart grid,” IEEE Communications Magazine, vol. 49, no. 4, pp. 53–59, 2011.

[2] N. Tekbiyik and E. Uysal-Biyikoglu, “Energy efficient wireless unicasting routing for vehicle-to-vehicle networks,” Journal of Network and Computer Applications, vol. 34, no. 5, pp. 1587–1614, 2011.

[3] D. Wang, L. Bai, X. Zhang et al., “Collaborative relay beamforming strategies for multiple destinations with guaranteed QoS in wireless machine-to-machine networks,” International Journal of Distributed Sensor Networks, vol. 2012, Article ID 525649, 10 pages, 2012.

[4] O. D. Incel, L. van Hoesel, P. Jansen, and P. Havinga, “MC-LMAC: a multi-channel MAC protocol for wireless sensor networks,” Ad Hoc Networks, vol. 9, no. 1, pp. 73–94, 2011.

[5] T. Shon, H. H. Choi, and E. J. Kim, “Joint routing and scheduling in multi-channel capillary machine-to-machine networks,” EURASIP Journal on Wireless Communications and Networking, vol. 2013, no. 1, pp. 1–9, 2013.

[6] J. So and N. H. Vaidya, “Multi-channel MAC for ad hoc networks: handling multi-channel hidden terminals using a single transceiver,” in Proceedings of the 5th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MoBiHoc ’04), pp. 222–233, May 2004.

[7] R. Draves, J. Padhye, and B. Zill, “Routing in multi-radio, multi-hop wireless mesh networks,” in Proceedings of the 10th Annual International Conference on Mobile Computing and Networking (MobiCom ’04), pp. 114–128, October 2004.

[8] R. Draves, J. Padhye, and B. Zill, “Comparison of routing metrics for static multi-hop wireless networks,” in Proceedings of the ACM SIGCOMM Computer Communication Review, vol. 34, pp. 133–144, September 2004.

[9] J. So and N. H. Vaidya, A Routing Protocol for Utilizing Multiple Channels in Multi-Hop Wireless Networks with a Single Transceiver, University of Illinois at Urbana-Champaign, 2004.

[10] D. B. Johnson, D. A. Maltz, and J. Broch, “DSR: the dynamic source routing protocol for multi-hop wireless ad hoc networks,” Ad Hoc Networking, vol. 5, pp. 139–172, 2001.

[11] D. S. J. De Couto, D. Aguayo, J. Bicket, and R. Morris, “A high-throughput path metric for multi-hop wireless routing,” Wireless Networks, vol. 11, no. 4, pp. 419–434, 2005.

[12] M. Mosko and J. J. Garcia, “Multipath routing in wireless mesh networks,” in Proceedings of the IEEE International Conference on Mobile Adhoc and Sensor Systems (MASS ’06), pp. 741–746, 2006.

[13] Z. Yuan, R. Yu, M. Zhao et al., “Vibration model based multipath traffic allocation algorithm for wireless mesh networks,” Journal of Information and Computational Science, vol. 10, no. 5, pp. 1253–1260, 2013.

[14] S. O. Reza Moheimani and A. J. Fleming, Piezoelectric Transducers for Vibration Control and Damping, Springer, 2010.

[15] A. D. Nashif, D. I. G. Jones, and J. P. Henderson, Vibration Damping, John Wiley and Sons, New York, NY, USA, 1985.

[16] Z.-G. Chen, F. Zeng, and Q.-H. Li, “Delay-constrained and jitter-optimized traffic allocation for multipath routing in wireless mesh networks,” Journal on Communications, vol. 32, no. 1, pp. 1–8, 2011.

[17] X. Chang, “Network simulations with OPNET,” in Proceedings of the Winter Simulation Conference Proceedings (WSC ’99), vol. I, pp. 307–316, December 1999.
[18] L. Qian, N. Song, and X. Li, "Detection of wormhole attacks in multi-path routed wireless ad hoc networks: a statistical analysis approach," *Journal of Network and Computer Applications*, vol. 30, no. 1, pp. 308–330, 2007.
