Technical Disclosure Commons

Defensive Publications Series

August 28, 2019

DYNAMICAL LINK METRIC ADJUSTMENT USING CLASSIFICATION AND REGRESSION TREE (CART) AND SOFTWARE DEFINED NETWORKING (SDN) TECHNOLOGIES

Lele Zhang

Yajun Xia

Jim Lu

Yinfang Wang

DYNAMICAL LINK METRIC ADJUSTMENT USING CLASSIFICATION AND REGRESSION TREE (CART) AND SOFTWARE DEFINED NETWORKING (SDN) TECHNOLOGIES

AUTHORS:

Lele Zhang
Yajun Xia
Jim Lu
Yinfang Wang

ABSTRACT

Presented herein are techniques to dynamically switch between different link metric algorithms based on Classification And Regression Tree (CART) and Software Defined Networking (SDN) technologies.

DETAILED DESCRIPTION

Low-power and Lossy Networks (LLNs) communicate using low data rate links (e.g., Institute of Electrical and Electronics Engineers (IEEE) 802.15.4g, IEEE 1901.2, *etc*.), which face many challenges. For example, LLNs communicate over a physical medium that is strongly affected by environmental conditions that change over time. Some examples include temporal changes in interference (e.g., other wireless networks or electrical appliances), physical obstructions (e.g., doors opening/closing or seasonal changes in foliage density of trees), and propagation characteristics of the physical media (e.g., temperature or humidity changes). The time scales of such temporal changes can range between milliseconds (e.g., transmissions from other transceivers) to months (e.g., seasonal changes of outdoor environment). Additionally, low-cost and low-power designs limit the capabilities of the transceiver. In particular, LLN transceivers typically support limited link margin, making the effects of interference and environmental changes visible to link and network protocols. Interference may be external (non-network devices generating electromagnetic interference) or internal (other network devices communicating within the same frequency band).

Over the past decade, certain vendors have provided Connected Grid Mesh networks (i.e., CG-Mesh) for Advanced Metering Infrastructure (AMI) and Distributed Automation (DA) Gateways customers as a LLN solution. CG-Mesh network is a tree-based topology wireless mesh network (WMN) per IPv6 routing protocol for a Low-power and Lossy network (RPL). However, CG-Mesh utilizes a link metric to form the topology through the WMN, such as expected transmission count (ETX). ETX is a measure of the quality of a path between two nodes in a wireless packet data network and it is used extensively in mesh networking algorithms. FIG. 1, below, illustrates the use of ETX in a network.

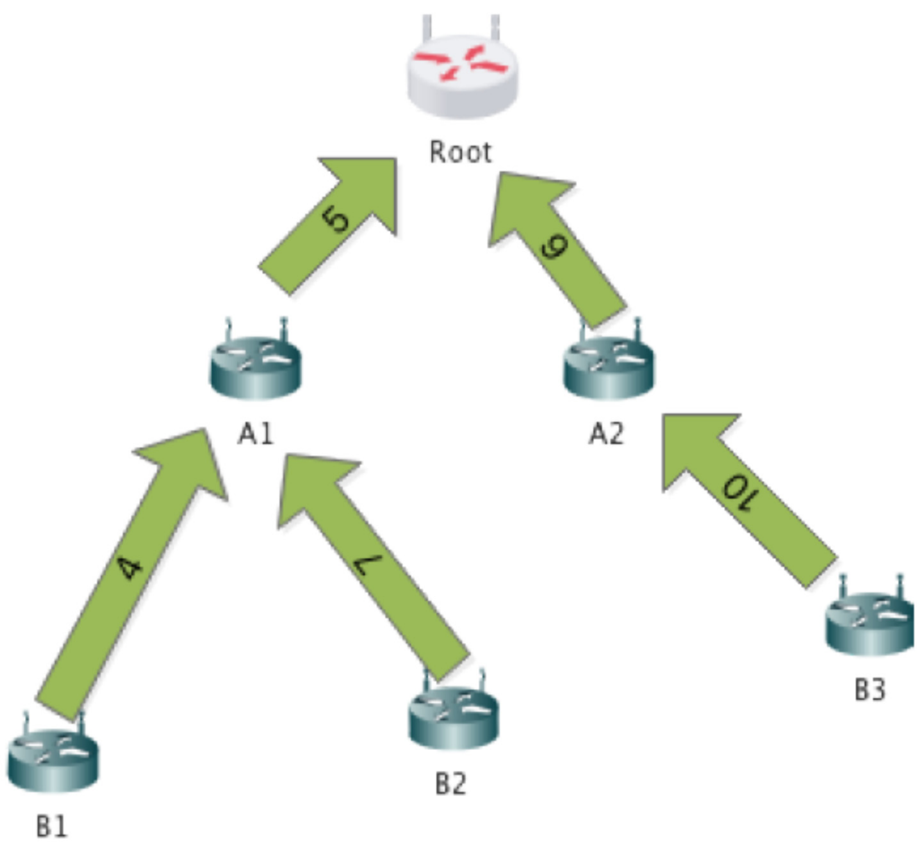

FIG. 1

In general, ETX is useful to most of scenarios. However, other existing metric algorithms are used for certain cases for which ETX is not useful. These existing metrics include:

1. ETX presents the probability that a packet successfully reaches the receiver, so the weights of both a long packet (i.e., payload more than 1000 bytes) and a small one (i.e., carries 50 bytes payload) are the same. However, in some cases, these times are not the same. As a result, the Expected Transmission Time (ETT) method is proposed to estimate the link metric considering the size of packet.
2. A device in WMN often has constrained resources (e.g., random access memory (RAM)). As such, there may not be enough buffers to queue received packets for each node. Based on this, Worse case ETX (W-ETX) is proposed to correct this defect.
3. Some devices in WMN, such as battery-powered devices, are power-limited. These devices should be considered in terms of remaining energy as the primary metric factor. As such, Expected Transmission Energy (ETE) is proposed.
4. Other link metric algorithms include, for example, Link Quality Level (LQI) based on link quality index, Packet Forwarding Indication (PFI), Link Latency (LL) and so on.

The above and other existing metrics have both advantages and drawbacks. In general, no existing metric is able to cover all practice use cases. For example, a first example, referred to as “Case 1” is shown below in FIG. 2.

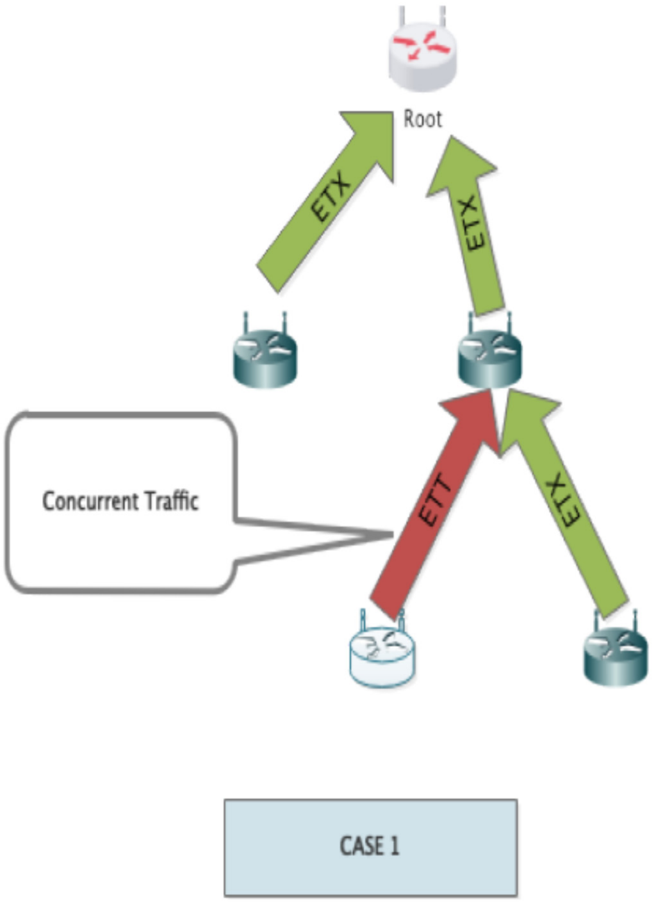

FIG. 2

In Case 1, a customer "I" needs to read a large amount of data from a bundle of nodes (e.g., ten) at 7 am. However, too much concurrent traffic may crash this block of the network, which leads to high packet loss in ratio while an existing link metric algorithm (like ETX) is fixed. However, if it is possible to switch the link metric algorithms dynamically, the network may be more stable.

A second example, referred to as "Case 2" is shown below in FIG. 3.

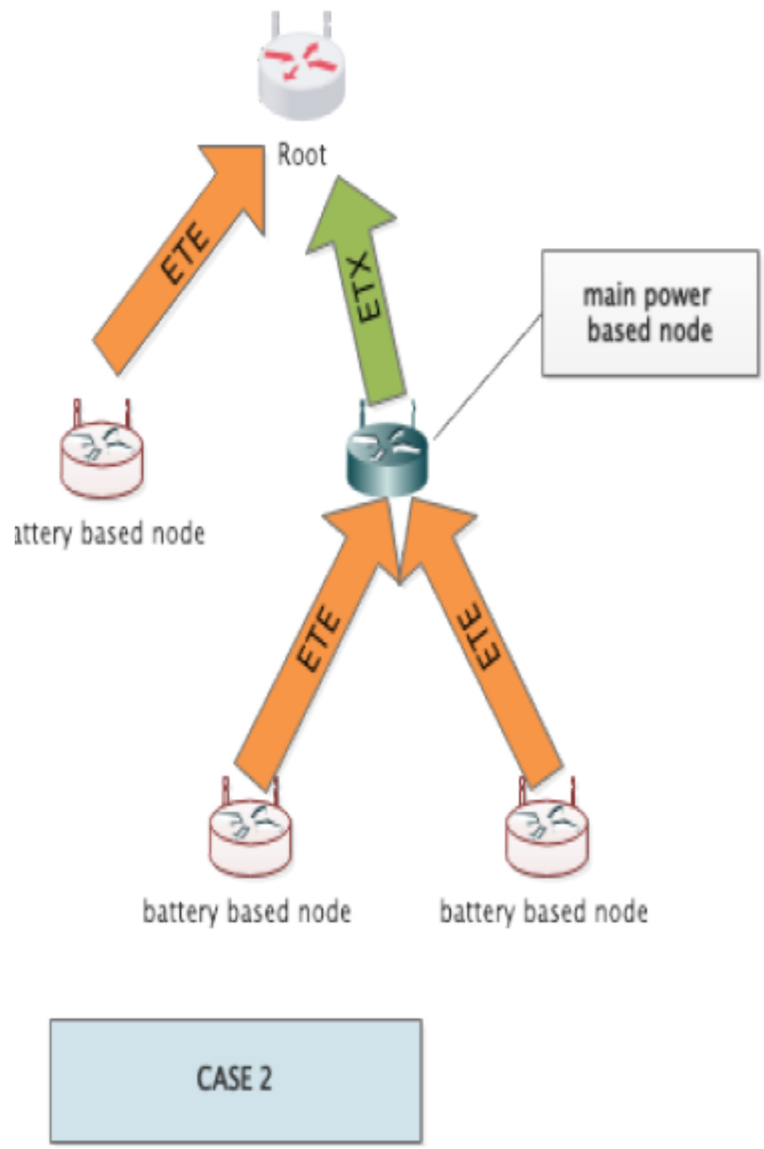

FIG. 3

In Case 2, a heterogeneous WMN exists with two kinds of units, one is powered by main power (e.g., AC or large backup battery), while the other is powered by a small battery (e.g., two triple-A battery). Therefore, these two difference units they could use different link metric algorithms to obtain better performance.

In view of the above, presented herein are techniques to dynamically switch utilized link metric algorithms based on Classification And Regression Tree (CART) and Software Defined Networking (SDN) technologies.

More specifically, CART is a popular kind of decision tree algorithm in machine learning (ML). CART is a binary-tree that supports both classification and regression problems. Since the problem presented herein is a classification problem, it is possible to use a GINI algorithm to generate the required DT, and then using pruning to get the most optimal subtree, as desired.

The classical CART algorithm is composed of two steps:

1. Decision Tree (DT) Generating: to produce a DT with the CART method and make sure this DT is as large as possible.
2. DT Pruning: In order to select the most optimal subtree, the DT needs to be pruned with a loss function (LF).

Therefore, based on the above the techniques presented herein are comprised of the following parts: (1) Generation of Classification Tree, (2) Pruning CART, and (3) use of cloud computing and SDN architecture to generate policies for each device in WMN.

<u>Part One: Generation of Classification Tree</u>

Link metric estimations are easily influenced by a number of factors that include, but are not limited to:

1. Device type: energy sensitive (battery based) or energy non-sensitive (main power) where generally speaking, the energy sensitive devices tend to adopt ETE algorithms and the energy non-sensitive devices tend to adopt ETX.
2. Latency between neighbors: latency sensitive devices prefer to adopt an LL method.
3. Traffic status: If there is congestion due to concurrent traffic, the W-ETX is recommended for better load balancing.
4. LQI value between neighbors: If there is strong interference among neighbors, then the LQL algorithm is better than other algorithms.
5. The free queue length: because most of WMN nodes are resource constrained, they are often short of usage queue buffers, so W-ETX is recommended at that time.

6. Temperature and humidity: Both of these factors are related to environment, thus the LQL algorithm may be more suitable under the worst case conditions.
7. Received signal strength indicator: similar to LQI, the LQL algorithm is preferred in some cases.
8. Phymode (e.g., 2FSK or OFDM), data rate (e.g., 50 Kbps or 150 Kbps, etc.)
9. The amount of children: the more children the node has, the stronger the competition exists among neighborhood.

When these factors are placed into a set "S," $S = \{s_1, s_2, ..., s_n\}$, where each trait s_i, may have several possible values. For example, if the first factor is a device type, there may be two possible values. Assuming that many samples for a training named set "D" are obtained, the samples can be divided into two subsets, where one corresponds to energy-sensitive devices and the other corresponds to non-energy-sensitive devices. Finally, it is possible to calculate the GINI index for set "D" as shown below in Equation 1.

Equation 1

$$Gini(D, s_i) = \frac{|D_1|}{|D|}Gini(D_1) + \frac{|D_2|}{|D|}Gini(D_2)$$

Going through all of traits in set S, the GINI values for all possible subsets can be determined. Principally, the division with the smallest GINI index for sample set D may be chosen. The above steps are repeated until a stop condition is reached. The stop conditions may include:

1. The amount of samples are less than predefined threshold.
2. The GINI index is less than the predefined threshold.
3. No more traits to analyze.

At last, the CART trees are obtained. In addition, the details of the GINI index may be given where it is a measurement between 0 and 1. In this case, 0 denotes 100% similarity and 1 denotes 0% similarity. If the gross samples contain more classes, then the GINI index will be bigger and have very similar with information entropy, which is another metric concept that comes from GBDT decision tree theory.

If the samples have "K" classes, and the probability of a class K is "pk," then the GINI formula is given as shown below in Equation 2.

Equation 2

$$Gini(p) = \sum_{k=1}^{k} p_k(1 - p_k) = 1 - \sum_{k=1}^{k} p_k^2$$

Therefore, the GINI value of sample set D is given as shown below in Equation 3.

Equation 3

$$Gini(D) = 1 - \sum_{k=1}^{k} \left(\frac{|C_k|}{|D|}\right)^2$$

In Equation 3, "C_k" denotes a subset which carries class k in sample set D. Thereby we could deduce the Equation 1, above.

Part Two: Pruning CART

ML solutions, such as DT, often face an over-fitting problem, which means the result look promising in a training set, but are terrible in practice. In order to overcome this problem, the techniques presented herein prune some subtrees to make the CART tree smaller. FIG. 4, below illustrates a tree "T," a first subtree "T_{t2}" and a second subtree "T-T_{t2}".

•

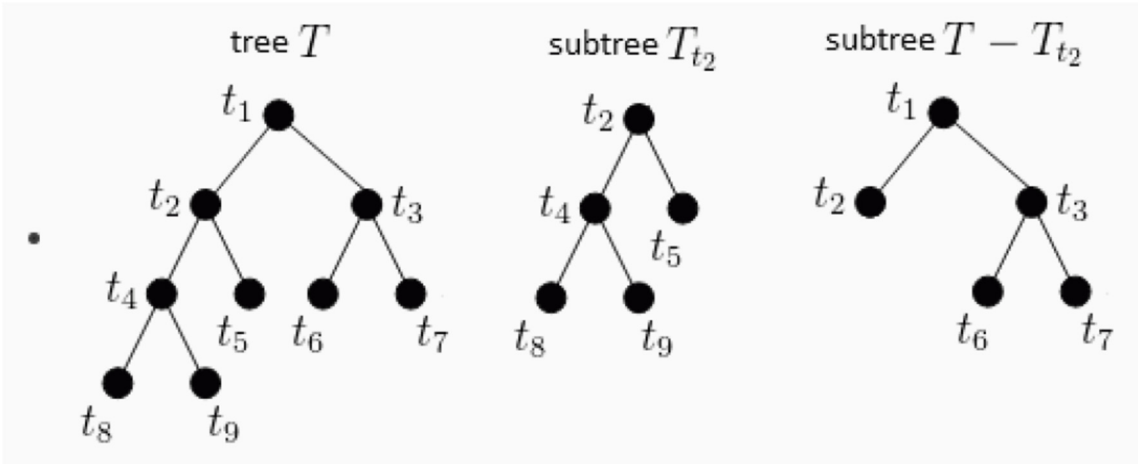

FIG. 4

There are many methods to fix over-fitting in a DT algorithm. Proposed herein is the use of the Cost-Complexity Pruning (CCP) for pruning CART. The core of CCP is loss function, which is described as below in Equation 4.

Equation 4

$$C_\alpha(T) = C(T) + \alpha|T|$$

In Equation 4, "C(T)" refers to deviation (e.g., GINI), "|T|" refers to a nodes' quantity, alpha (no less than 0) is a parameter which takes in charge of balance the grade of fitting based on the CART algorithm. Unlike other DT algorithms, such as ID3 or C4.5, the parameter alpha is not given by human input. Instead, alpha is determined by samples, which is more suitable in the techniques presented herein.

<u>Part Three: Use of cloud computing and SDN architecture to generate policies for each device in WMN.</u>

As is known, the common devices in WMN are constrained with resources, such as memory or computing ability. As such, it is not practicable (or possible) for a device to run a CART result by itself. However, it is possible to run CART in a cloud computing environment. As such, proposed here is the use of cloud computing and the SDN architecture for generation of CART solutions. This arrangement is shown below in FIG. 5.

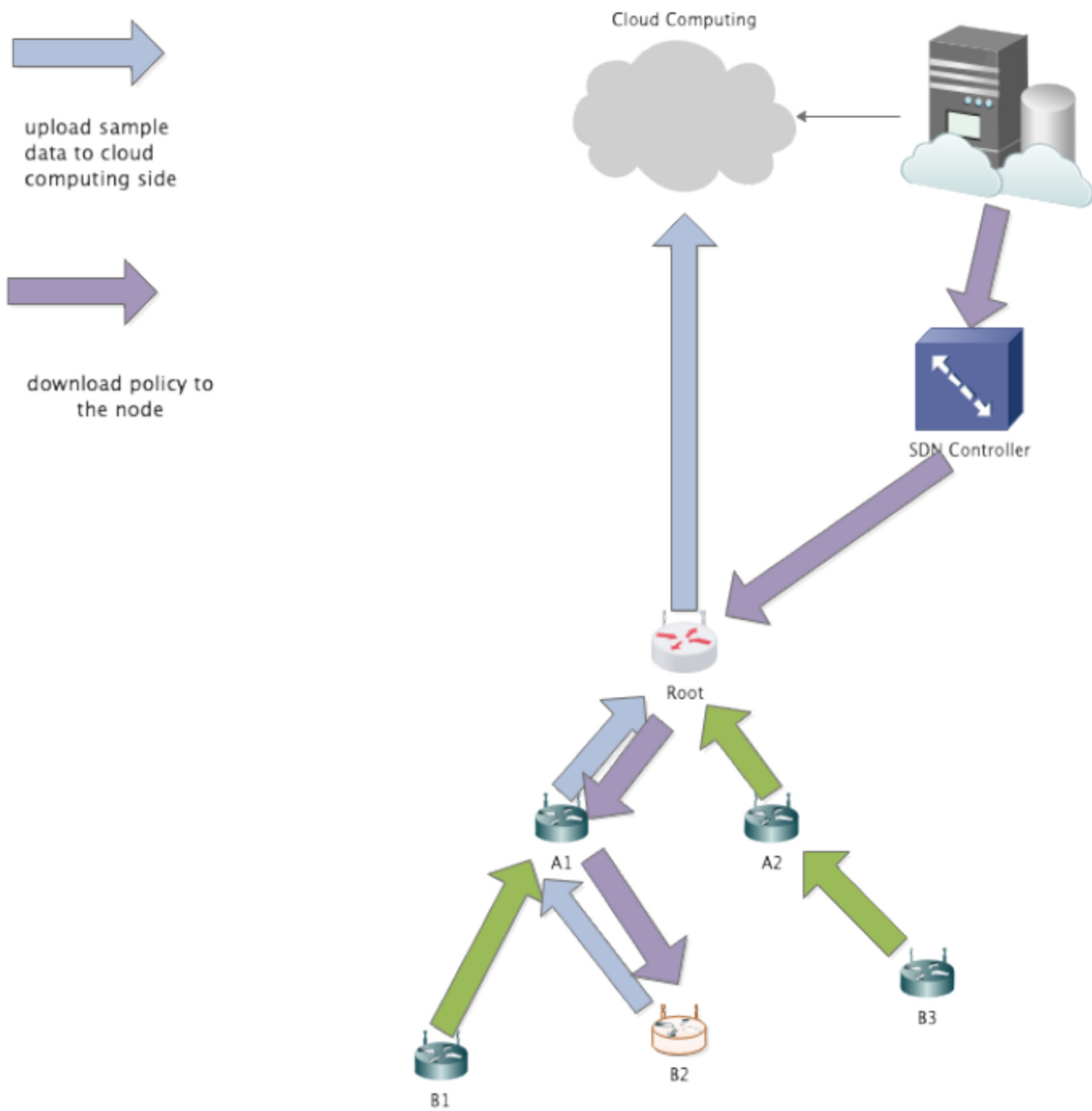

FIG. 5

As shown in FIG. 5, node B2 collects data according to the factors listed above in "Part One," which are then uploaded to the cloud computing environment. The cloud computing environment calculates out the CART tree for node B2 using the samples, and then provides the result/answer to the SDN controller. The SDN controller generates the

CART policy for node B2 to adjust its link metric algorithms dynamically based on the real-time situation. In the meantime, node B2 continues to upload samples to the cloud computing environment periodically (e.g., maybe 24 hours per round). As a result, the cloud computing environment can adjust the CART result based on the latest received samples, and then update with SDN controller. Therefore, the SDN controller could update to a new policy for node B2, as determined based on the latest samples. The total process is shown below in FIG. 6.

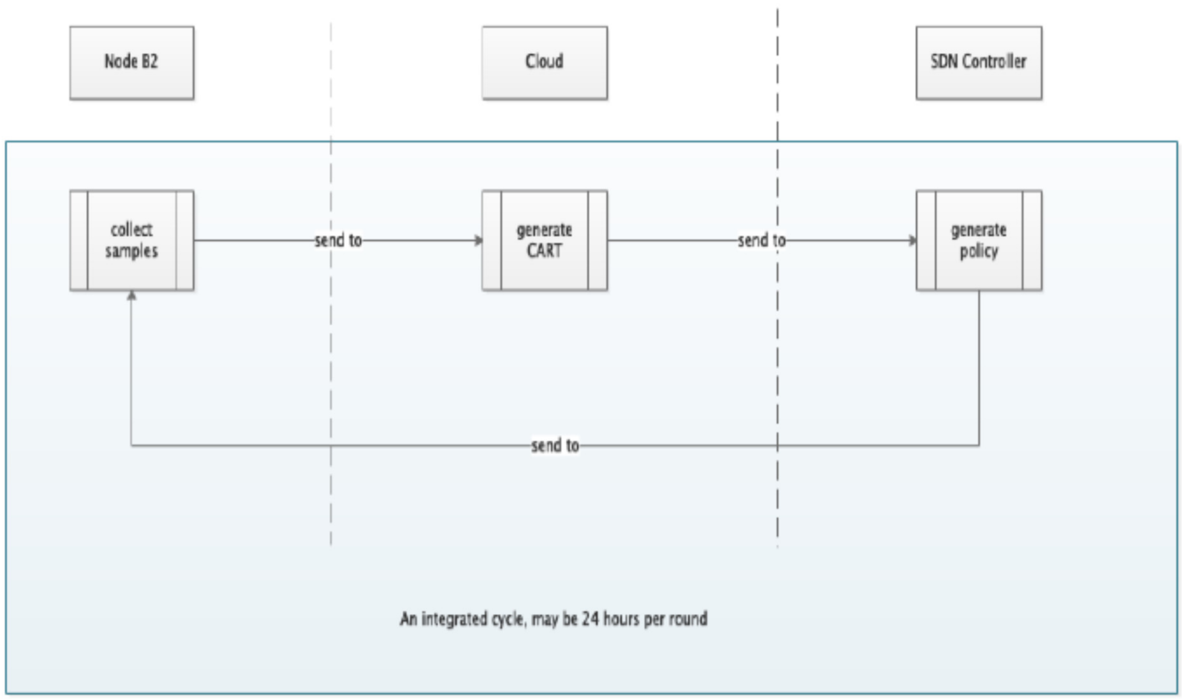

FIG. 6

As detailed above, the WMN has conventionally adopted only one link metric algorithm for use in an instance (i.e., a bundle of wireless nodes), although it could not satisfy all the requirements of devices in the PAN. Therefore, presented herein is the use of a ML method, known as CART, to dynamically apply mixed link metric algorithms. As detailed above, the techniques presented herein also utilize SDN and cloud computing architectures. In general, the techniques presented herein provide the most optimal metric solution for all nodes in a WMN, which could improve the performance of the WMN, make the WMN it more stable, and promote the throughput in the presence of concurrent traffic.