Cost-efficient QoS-Aware Data Acquisition Point Placement for Advanced Metering Infrastructure

Fariba Aalamifar and Lutz Lampe

Department of Electrical and Computer Engineering, The University of British Columbia, {faribaa, lampe}@ece.ubc.ca

Abstract

In an advanced metering infrastructure (AMI), data acquisition points (DAPs) are responsible for collecting traffic from several smart meters and automated devices and transmitting them to the utility control center. Although the problem of optimized data collector placement has already been addressed for wireless broadband and sensor networks, DAP placement is quite a new research area for AMIs. In this paper, we investigate the minimum required number of DAPs and their optimized locations on top of the existing utility poles in a distribution grid such that smart grid quality of service requirements can best be provided. In order to solve the problem for large-scale AMIs, we devise a novel heuristic algorithm using a greedy approach for identifying potential pole locations for DAP placement and the Dijkstra's shortest path algorithm for constructing reliable routes. We employ the characteristics of medium access schemes from the IEEE 802.15.4g smart utility network (SUN) standard, and consider mission-critical and non-critical smart grid traffic. The performance and time-complexity of our algorithm are compared with those obtained by the IBM CPLEX software for small scenarios. Finally, we apply our devised DAP placement algorithm to examples of realistic smart grid AMI topologies.

I. INTRODUCTION

As part of the broader concept of smart grids, advanced metering infrastructures (AMIs) are being massively deployed almost everywhere in the world. AMIs are responsible for reading the energy consumption from thousands of smart meters (SMs) [1]–[4], monitor the last-mile automated devices for reporting emergency events [3], [4] such as electricity blackout and also unauthorized access to the power system [5]. In an AMI, due to the large number of devices and their distances, data collectors are installed to collect traffic from several endpoints and transmit them to the utility control center on their behalf.

The placement of collector nodes, which are known as data acquisition points (DAPs) [4], [6] or aggregators [7], [8] in smart grid communication networks (SGCNs) and as relay station [9], gateway [10] or sink [11], [12] in broadband wireless access networks and sensor networks, respectively, has previously been investigated [13]–[16]. However, there is a combination of features and requirements in AMI that render the problem sufficiently different from the data collector placement in other types of networks so that a new problem formulation and solution for network planning are needed. For example, in sensor networks, the collector nodes can be placed on selected endpoint nodes [10] or in arbitrary locations [13]. Different from this, in a distribution grid with overhead powerlines, the utility poles are ideal locations for DAP placement [4], since this extends network coverage and also eliminates the cost of new tower installations. Moreover, since the locations of utility poles are determined based on the power grid infrastructure, for example they are often located along roads and thus not uniformly distributed in a coverage area, it is not straightforward to apply the existing placement algorithms to place DAPs in AMIs. Another major difference is that the on-time delivery of smart grid traffic to the utility control center and automated devices is critical for the correct operation of the electrical power grid [3], [17]. Also, due to the existence of two types of traffic classes namely, mission-critical and non-critical traffic, different scheduling schemes should be employed so that the

quality of service (QoS) associated with both traffic can be maintained. In addition, due to the existence of rural areas, a multi-hop communication infrastructure is required in order to access further nodes.

Accordingly, the main design considerations for the placement of collector nodes in AMIs are the number and location of DAPs so that 1) the network coverage is ensured, 2) the required reliabilities associated with different types of smart grid traffic classes are satisfied, and 3) existing infrastructures (utility poles) are used. Thereby, two types of access architectures from automated devices to DAPs are possible: a) direct and b) multi-hop communication. In this paper, we address the multi-hop connectivity case as it allows for accessing more remote devices and requires a smaller number of DAPs.

The mathematical optimization formulation for DAP placement on top of existing utility poles is an integer programming (IP) problem and is NP-hard. For cases with small number of nodes, say no more than 200, the IBM CPLEX software [18] and the GLPK solver [19] are typically used for finding optimized node locations. However, for cases with notably larger number of nodes, a heuristic algorithm needs to be developed [14], [20], [21].

Heuristic algorithms proposed for relay placement are typically based on cover-set or facility-location algorithms. For example, references [10] and [22] propose weighted cover-set algorithms for respectively gateway and reader placement for wireless sensors and radio-frequency identification nodes. Reference [21] applies the minimum-cover-set algorithm for finding the optimal location of DAPs for both single and multi-hop access in SGCNs. When the network becomes large, their heuristic algorithm breaks the area into smaller squares which can be handled by the optimizer. Their post-optimization step involves merging the solution of smaller squares by removing the redundant poles located in square edges. This step of their heuristic algorithm has a high complexity, because every pole that is not selected is checked to see if it can replace a subset of two or more selected poles. In our preliminary work [23], we have proposed a modified K-means algorithm for DAP placement in the single-hop communication scenario only

considering network coverage, assuming SMs and poles are uniformly distributed through the area. The K-means algorithm chooses random locations as primary potential locations for DAP placement and all the network construction is conducted based on these locations. These random locations are eventually mapped to the closest pole. However, there is a higher possibility that such a mapping would result in the violation of QoS constraints when a realistic data set is considered, for example when poles are aligned with the road structure. Therefore, in this paper, we apply a different and more suitable heuristic algorithm by which the network is constructed from pole locations. In [20], the authors develop a K-means based algorithm for placing a fixed number of aggregators on selected utility poles with the objective of minimizing the total number of hops SMs require to access the selected data aggregators. This work is among the first to consider multi-hop communication and minimize the experienced delays by minimizing the total number of hops. However, limiting the number of hops only addresses the effect of transmission delay and ignores the effect of congestion delay which explicitly depends on the number of competitors and their arrival rates at each hop.

References [24] and [25] propose aggregator placement solutions for respectively maintaining and maximizing the obtained QoS in an AMI. They use M/D/1 and M/G/1 queuing models for computing the expected latency over the designed infrastructure. However, the mission-critical and non-critical smart grid traffic need the guarantees of certain latency requirements with certain probabilities (i.e., ensuring certain reliabilities), which is not provided through the solutions in [24], [25].

In this paper, we do not adopt the average latency model with fixed or minimum number of hops criteria considered in [20], [21], [24], [25]. Instead, to meet latency requirements of smart grid traffic, in Section II, we compute the probability of achieving a certain latency requirement for both mission-critical and non-critical traffic. To this end, we employ the IEEE 802.15.4g MAC protocol [26] with the contention-free-period (CFP) and the contention-access-period (CAP) for scheduling critical and non-critical smart grid traffic. Then, we devise an optimization problem

in Section II and propose a novel heuristic algorithm for solving the problem in Section III. The heuristic algorithm approximates the minimum required number of DAPs through the use of a greedy algorithm for selecting potential pole locations for aggregator placement. In order to connect nodes through reliable routes, we use the Dijkstra algorithm for identifying transmission paths with the maximum packet success ratio. In Section IV, we provide performance results based on realistic locations for SMs and poles, which we have obtained from BC Hydro, a Canadian utility in the province of British Columbia. The results show that the paths found by our algorithm satisfy the latency requirements for both types of traffic to a specified level. We also compare the optimality and complexity of our solution for small-scale scenarios with the branch and cut algorithm offered by the IBM CPLEX software [18]. Finally, we conclude the paper in Section V.

II. SYSTEM MODEL AND PROBLEM FORMULATION

We consider a distribution grid with overhead power lines suspended from utility poles delivering electricity to homes or businesses equipped with SMs. Some utility poles host DAPs, each of which is wirelessly connected to a subset of the endpoints (SMs) either in a single-hop or multi-hop manner. The multi-hop communication utilizes IEEE 802.15.4g [26] for connecting SMs to each other or to the DAPs. We also assume that the following types of traffic, as listed in Table I, are passing through the grid.

1) Non-critical (NC) traffic such as reading the home energy consumption, periodically or on-demand.

2) Mission-critical (MC) traffic such as alert notifications, including meter tampering and power theft, remote control commands, and power quality (e.g., voltage, phase or current) notifications [28]. The MC traffic is usually modelled according to a Poisson process [24], [29].

According to the OpenSG Forum [3], reliability is defined as the probability that a packet can

TABLE I

MISSION CRITICAL AND NON-CRITICAL TRAFFIC PROPERTIES [27].

Traffic Class	Traffic Name	Packet Size (Bytes)	Arrival Frequency	Traffic Type	Required Latency
NC	Periodic Meter Reading (MR)	250	15 min	Deterministic	5 sec
NC	On-demand MR Request	50	5 days	Poisson	30 sec
NC	On-demand MR Response Data	250	5 days	Poisson	30 sec
MC	Power Quality Notifications	100	5 min	Poisson	1 sec
MC	Remote Control Commands	100	1 day	Poisson	1 sec
MC	Alert Notifications	50	1 week	Poisson	3 sec

successfully be received at the destination within its required latency. Therefore, in order to meet the reliability requirements of the smart grid traffic, both the route quality in terms of the packet success rate and the probability of exceeding the latency requirement over the route should be taken into account. We formulate the link quality in Section II-A and the probability of latency satisfaction for NC and MC traffic in Section II-B. Using these expressions, we formulate the obtained reliability over a certain route in Section II-C.

A. Link Quality

The link quality, defined as the probability of a successful packet transmission on the link between nodes i and j, is obtained as

$$1 - \epsilon_{ij} = 1 - \mathcal{Q}(\gamma_{ij}), \tag{1}$$

where ϵ_{ij} is the link packet error rate (PER), γ_{ij} is the signal to interference and noise ratio (SINR) and $\mathcal{Q}$ maps the SINR to the PER based on the modulation and coding scheme. The SINR is given by

$$\gamma_{ij} = \frac{P_{\text{tx}}}{(N_0' + I) \ PL(d_{ij}) \ \eta \ \delta} \tag{2}$$

where P_{tx} is the transmit power, PL is the distance-dependent path loss, the variable d_{ij} denotes the distance between nodes i and j, η is the fading margin, and $N_0' = N_0 F$ where N_0 and F are respectively the receiver noise power spectral density and noise factor. The variable I denotes the interference, which accounts for the inter-operator interference when operating in the unlicensed band or when the same block of frequency is used by other operators or applications as well as cell-to-cell interference [30]. Furthermore, δ is the penetration loss which is present when SMs are located inside the building. The pathloss component $PL(d_{ij})$ depends on the area type. According to the NIST PAP2 guideline [30], the Erceg SUI propagation model best emulates the channel propagation for rural and suburban scenarios. For urban areas, the ITU-R M.2135-1 (outdoor) and ITU-R M.1225 (indoor) propagation models are suggested.

B. Delay Model

The IEEE 802.15.4g MAC protocol provides two types of medium access periods, namely CFP and CAP, within each frame. A node stores the MC and NC traffic in different queues, and schedules the mission-critical traffic through the CFPs using the time division multiple access (TDMA) scheme, and the non-critical traffic within the CAP time slots using the carrier sensing multiple access/collision avoidance (CSMA/CA) scheme. We hereafter denote the number of available time slots per frame in the CFP and CAP by N_{T} and N_{C}, respectively.

Let us assume that the traffic from each node should be received at the destination within a time period of L seconds. In order to compute the probability that an NC or MC packet can be transmitted within this delay requirement, we need to translate L to its equivalent number of available slots via

$$N_{\mathrm{s}} = (\text{MC or NC}) = \frac{L}{T_{\mathrm{F}}} \times N_{\mathrm{T}} \text{ or } N_{\mathrm{C}}, \tag{3}$$

where T_{F} is the frame duration in seconds. As we are dealing with a multihop communication system, the cumulative waiting time during all the hops should be less than the required latency.

Let us assume node n is located at depth H_n of the network and r_{hn} is the relay node which forwards the message of node n at hop h where $1 \leq h \leq H_n$. To meet the required delay for node n we allow

$$S = \left\lfloor \frac{N_{\mathrm{s}}}{H_n} \right\rfloor \tag{4}$$

time slots to be consumed at each of its forwarding nodes. This conservative assumption allows us to guarantee the required reliability. It should be noted that in practice, a larger delay may be consumed at some hop, while the total delay is still maintained. We hereafter assume each packet, even the largest-size packets of 250 bytes, can be transmitted within one time slot.

There are several components included in the total packet delay, namely transmission, queuing, medium access, and propagation delay. Propagation delay is usually ignored for links with short distances [30]. In the following, we first compute the average queuing delay. We then formulate the latency requirement that should be met for QoS satisfaction at each hop by deducting the queuing and transmission delay from the total allowed delay. Next, we mathematically derive the probability of meeting this required delay based on the MAC protocol specifications of the 802.15.4g standard.

1) Queuing Delay: For tractability of computing the queuing delay, we assume all sources generate Poisson traffic, which has been shown to be a sufficiently accurate approximation for mixed traffic as considered in our work [24]. We further assume that the Poisson traffic model also applies to nodes forwarding packets, which is justified if the traffic load at each node is low [31]–[34] and will also be verified numerically in Section IV-C for typical traffic scenarios of our application. We then accordingly apply the M/G/1 queueing model in order to compute the average waiting time at the queue of node x [31], [32]. According to the Pollaczek-Khinchin formula [35], the waiting time in time slots is given by

$$T_{\mathrm{Q}_x} = \frac{\lambda_x E[Y_x^2]}{2(1 - \frac{\lambda_x}{\mu_x})} \tag{5}$$

where

$$\lambda_x = \sigma_x \lambda_0 (N_{\text{f}x} + 1) \tag{6}$$

is the aggregated arrival rate at the node, $N_{\text{f}x}$ denotes the total number of feeding nodes that are directly or indirectly connected to node x, λ_0 is the average traffic generation rate per node, and σ_x gives the expected number of times that the packet should be re-transmitted, which will be calculated later in this section. μ_x is the packet service rate and $E[Y_x^2]$ denotes the second moment of the service time for both NC and MC traffic, which is given by

$$E[Y_x^2] = \frac{N_{\text{C}} + N_{\text{T}}}{N_{\text{C}} \text{ or } N_{\text{T}}} \sum_{k=1}^{S} \Big(R_x(k) - R_x(k-1) \Big) k^2, \tag{7}$$

where $R_x(k)$ is the probability that the packet can successfully be transmitted within k CAP or CFP slots. Variables μ_x and $R_x(k)$ are obtained later in this section.

2) *Medium Access Delay:* Consider that r_{hn} has $N_{r_{hn}}$ neighbours, which we collect in the set $\Psi_{r_{hn}}$, and let $\mathcal{P}_{r_{hn}} = \{p_x : x \in \Psi_{r_{hn}}\}$ be the probabilities that these neighbours have a packet for transmission, given by $p_x = \frac{\lambda_x}{\mu_x}$ [35], where λ_x has been defined in (6) above, and μ_x is the service rate. μ_x is obtained later in the following section.

Here, we describe how the probability of exceeding a certain delay is computed for the traffic generated by node n for the above-mentioned scheduling schemes as a function of S and $\mathcal{P}_{r_{hn}}$. In order to increase the obtained reliability, for each packet, we allow up to N_{ARQ} transmission attempts.

- **Non-critical traffic:** Under the slotted CSMA/CA model, each node with the NC traffic, at each transmission attempt, would sense the channel at most $M + 1$ times. At each sensing stage $m = 0, 1, \cdots M$, it selects a random time slot within the backoff window, W_m, with equal probability. According to the IEEE 802.15.4g standard, in slotted CSMA/CA model, each node should identify the channel as idle for two consecutive slots before changing to transmission mode. If two nodes sense the channel as idle at the same time, there would be a collision. We note that since the length of CAP is comparable to the average size of
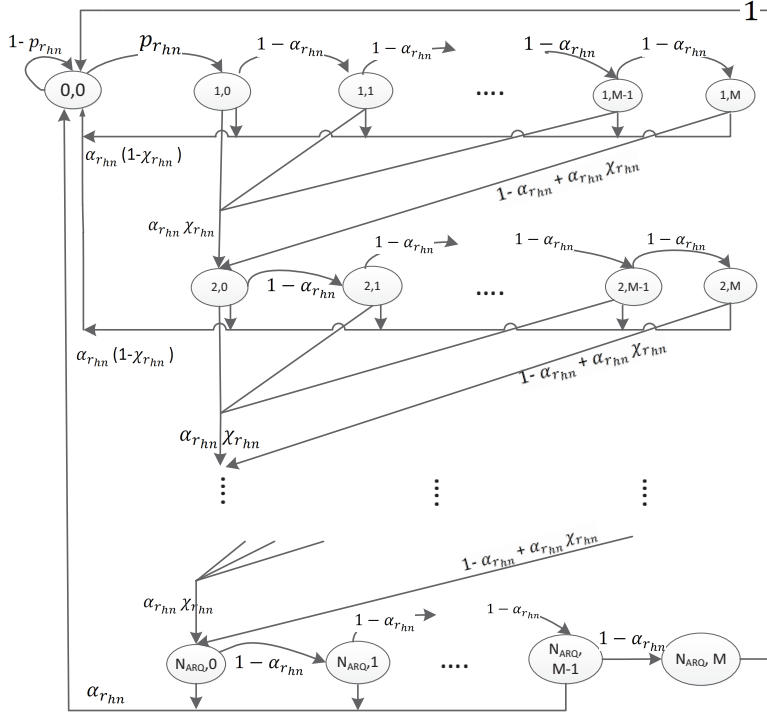
Fig. 1. Markov chain for the CSMA/CA process. State (i, m), $1 \leq i \leq N_{\mathrm{ARQ}}$, $0 \leq m \leq M$ represents the sensing stage m in the ith transmission attempt, and $(0, 0)$ is the state of having no packets for transmission. $p_{r_{hn}}$ is the probability that the node has a packet for transmission, $\alpha_{r_{hn}}$ is the probability that the channel is idle and $1 - \chi_{r_{hn}}$ is the probability that the packet has successfully been transmitted.

a backoff stage, the accumulated traffic during CFP would be uniformly distributed over the CAP and therefore, similar to [32], [36], [37], which consider inactive periods between CAPs, the probability that the channel is idle is assumed to be constant within the CAP. Figure 1 shows the Markov chain model associated with the CSMA/CA procedure. We define $\beta_{1_{r_{hn}}}$ as the probability that the channel is busy when sensing for the first time, $\beta_{2_{r_{hn}}}$ as the probability that the channel is busy when sensing for the second time, provided that the channel was idle for the first time, and

$$\alpha_{r_{hn}} = (1 - \beta_{1_{r_{hn}}})(1 - \beta_{2_{r_{hn}}}) \tag{8}$$

as the probability that the channel is determined as idle for two consecutive time slots. The

channel is determined as busy if the channel was idle for two consecutive time slots and at least a node has sensed the channel in those slots. Hence, the probability of $\beta_{1_{r_{hn}}}$ is obtained from [36]

$$\beta_{1_{r_{hn}}} = (1 - \beta_{1_{r_{hn}}})(1 - \beta_{2_{r_{hn}}}) \left(1 - \left(\prod_{x \in \Psi_{r_{hn}}} (1 - \xi_x) \right) \right), \tag{9}$$

where ξ_x is the probability that a neighbour node conducts its first carrier sensing attempt in an arbitrary time slot. The probability that the channel is determined as busy when sensing for the second time, given that the channel was idle for the first time is obtained from [36]

$$\beta_{2_{r_{hn}}} = (1 - \beta_{2_{r_{hn}}}) \left(1 - \left(\prod_{x \in \Psi_{r_{hn}}} (1 - \xi_x) \right) \right). \tag{10}$$

In order to compute ξ_x, we use the stationary probabilities associated with the Markov chain shown in Figure 1. Let $\boldsymbol{\pi}$ and $\mathbf{T}$ respectively denote the stationary distribution vector and transition matrix of this Markov chain. Solving the stationary state equation $\boldsymbol{\pi}\mathbf{T} = \boldsymbol{\pi}$ subject to $\sum_j \boldsymbol{\pi}_j = 1$, we can compute the probability of conducting the first carrier sensing attempt in an arbitrary time slot by a neighbour node as

$$\xi_x = \sum_{i=1}^{N_{\text{ARQ}}} \sum_{m=0}^{M} \frac{\boldsymbol{\pi}_{g(i,m)}}{W_m}, \tag{11}$$

where $g(i,m) = (i-1)(M+1) + m + 1$, $\boldsymbol{\pi}_{g(i,m)}$ is the probability of being in sensing stage m in transmission attempt i, and $\frac{1}{W_m}$ gives the probability of conducting the first carrier sensing attempt in an arbitrary time slot in stage m.

In order to compute the probability that a node can transmit its packet within the required latency, we need to compute the probability that the node senses the channel within the latency and also the channel is idle. Let us define $\theta_{r_{hn}}(k)$ as the probability that node r_{hn} senses the channel in time slot k and also the channel is idle. Since slot k can be sensed at any of the N_{ARQ} transmission attempts and $M+1$ backoff stages, $\theta_{r_{hn}}(k)$ is obtained as

$$\theta_{r_{hn}}(k) = \sum_{i=1}^{N_{\text{ARQ}}} \sum_{m=0}^{M} \zeta_{r_{hn}}(k,i,m) \, \alpha_{r_{hn}}, \tag{12}$$

where $\zeta_{r_{hn}}(k, i, m)$ is the probability of sensing the channel at slot k, in sensing stage m, in transmission attempt i. The variable $\zeta_{r_{hn}}(k, i, m)$ is computed based on the probability of having an unsuccessful transmission attempt (due to either finding the channel as busy during all $M+1$ backoff stages or due to packet transmission failure) in one of the previous d slots in the previous try and then sensing the channel at slot $k - d - 2$ in the current try,

$$\zeta_{r_{hn}}(k, i, m) = \begin{cases} \sum_{d=3(i-2)+1}^{k-2} \sum_{m'=0}^{M} \zeta_{r_{hn}}(d, i-1, m') \, \Delta_{m'} \, \phi(k-d-2, m), & i > 1, \\ \phi(k, m), & i = 1, \end{cases} \tag{13}$$

where at least 3 slots are consumed at each attempt (2 slots for sensing and 1 for transmission),

$$\Delta_{m'} = \begin{cases} \alpha_{r_{hn}} \chi_{r_{hn}}, & m' < M, \\ \alpha_{r_{hn}} \chi_{r_{hn}} + (1 - \alpha_{r_{hn}}), & m' = M, \end{cases}$$

and $\phi(k, m)$ is the probability of assessing the channel at slot k in sensing stage m. The value of $\phi(k, m)$ is also recursively computed as a cumulative probability of sensing the channel at slot j in the previous sensing stage, finding the channel as busy in either the first or second slot and accordingly, backing off for $k - j$ slots with probability $\frac{1}{W_m}$ in the current sensing stage m [38]. In other words, $\phi(k, m)$ can be calculated as

$$\phi(k, m) = \begin{cases} \sum_{j=1}^{k-1} \phi(j, m-1) \, \beta_1 \, \frac{1}{W_m} + \sum_{j=1}^{k-2} \phi(j, m-1) \, (1 - \beta_1) \, \beta_2 \, \frac{1}{W_m}, & m \geq 1, \ k \geq 1, \\ \frac{1}{W_0}, & m = 0, \quad k \geq 1, \\ 0, & k < 1. \end{cases} \tag{14}$$

Finally using (12) with (13)-(14) and (8)-(11), the probability that node r_{hn} can successfully transmit the packet within the required latency is obtained as

$$R_{r_{hn}}(S) = \sum_{k=1}^{S-T_{\mathrm{Q}}-1} \theta_{r_{hn}}(k)(1 - \chi_{r_{hn}}), \tag{15}$$

where $1 - \chi_{r_{hn}}$ is the probability that the packet can successfully be transmitted, i.e., the packet transmission does not fail due to a collision (given that the channel is determined

as idle, at least one other node senses the channel at the same time as r_{hn}) or due to a link error. It is obtained as

$$1 - \chi_{r_{hn}} = (1 - \epsilon_h) \left(\prod_{x \in \Psi_{r_{hn}}} (1 - \xi_x) \right), \tag{16}$$

where ϵ_h is the link PER between r_{hn} and the relay node at the next hop as defined in (1).

- **Mission-critical traffic:** In this section, we compute the probability that all the bandwidth requests from the neighbour nodes, can be scheduled within the latency requirement. According to [30], this probability is computed as

$$\Pr(\ell_{r_{hn}} \leq S) = \sum_{i=0}^{S-1} \left(\sum_{\psi \in \Psi_{r_{hn},i}} \prod_{j \in \psi} p_j \prod_{k \in \Psi_{r_{hn}} \backslash \psi} (1 - p_k) \right), \tag{17}$$

where $\ell_{r_{hn}}$ is the experienced delay at relay node r_{hn} over one transmission attempt, $\Psi_{r_{hn},i}$ is the set of all subsets of $\Psi_{r_{hn}}$ with size i. For Poisson traffic assumed here, the expression in (17) has the closed-form solution [39]

$$\Pr(\ell_{r_{hn}} \leq S) = \sum_{i=0}^{S-1} \frac{1}{N_{r_{hn}}+1} \sum_{\kappa=0}^{N_{r_{hn}}} e^{j\frac{-2\pi\kappa i}{N_{r_{hn}}+1}} \prod_{k=1}^{N_{r_{hn}}} \left(p_k e^{j\frac{2\pi\kappa}{N_{r_{hn}}+1}} + (1 - p_k) \right), \tag{18}$$

where j is the imaginary unit. Let us define $\mathcal{L}_{r_{hn},i}$ as the cumulative sum of delays over i transmission attempts. We can compute the obtained reliability at hop h after N_{ARQ} transmission attempts as

$$R_{r_{hn}}(S) = \sum_{i=1}^{N_{\mathrm{ARQ}}} \Pr(\mathcal{L}_{r_{hn},i} \leq S - T_{\mathrm{Q}}) \, (\epsilon_h)^{i-1} \, (1 - \epsilon_h), \tag{19}$$

where similar to the NC traffic, the probability of latency satisfaction at each attempt can be recursively computed based on the time that has elapsed in the previous attempts, i.e.,

$$\Pr(\mathcal{L}_{r_{hn},i} \leq S) = \sum_{k=i-1}^{S-1} \Pr(\mathcal{L}_{r_{hn},i-1} = k) \Pr(\ell_{r_{hn}} \leq S - k), i > 1, \tag{20}$$

where

$$\Pr(\mathcal{L}_{r_{hn},i} = k) = \sum_{d=i-1}^{k-1} \Pr(\mathcal{L}_{r_{hn},i-1} = d) \Pr(\ell_{r_{hn}} = k - d), i > 1, \tag{21}$$

and

$$\Pr(\ell_{r_{hn}} = u) = \frac{1}{N_{r_{hn}} + 1} \sum_{\kappa=0}^{N_{r_{hn}}} \mathrm{e}^{\mathrm{j}\frac{-2\pi\kappa(u-1)}{N_{r_{hn}}+1}} \prod_{k=1}^{N_{r_{hn}}} \left(p_k \mathrm{e}^{\mathrm{j}\frac{2\pi\kappa}{N_{r_{hn}}+1}} + (1 - p_k) \right). \qquad (22)$$

- **Computing service rates:** As mentioned earlier, in order to compute p_x, we need to compute the average service rate for the NC and MC traffic for node x. The average service rate for node x can be obtained as $\mu_x = \frac{1}{E[Y_x]}$, where $E[Y_x]$ is the mean packet service time, which is calculated as

$$E[Y_x] = \begin{cases} \frac{1}{N_C + N_T} \sum_{i=1}^{N_T} i + \sum_{m=0}^{M} (1 - \alpha_x)^m \frac{W_m + 2}{2} + \sum_{m=1}^{M} (1 - \alpha_x)^m \left(\frac{W_m + 2}{2N_C} N_T \right) + 1, & \text{CSMA/CA,} \\[2em] \frac{1}{N_C + N_T} \sum_{i=1}^{N_C} i + \left\lceil \frac{\frac{1}{2} \sum_{x' \in \Psi_x \cup x} \lambda_{x'} \frac{L}{H_x}}{N_T} \right\rceil (N_T + N_C) + \mathrm{mod}\,(\frac{1}{2} \sum_{x' \in \Psi_x \cup x} \lambda_{x'} \frac{L}{H_x}, N_T), & \text{TDMA,} \end{cases} \qquad (23)$$

that is, for the NC traffic $E[Y_x]$ is computed based on whether the packet has arrived during the CFP and accordingly, the corresponding CFP duration should be added to the service time. Also, we need to consider the expected time that is needed for backoff, plus adding another CFP if the channel is busy and the remaining CAP slots are not sufficient for a new backoff. Finally, one time slot is added for packet transmission. For the MC traffic, $E[Y_x]$ is computed based on whether the packet has arrived during the CAP and accordingly, the corresponding CAP duration should be added to the service time. Also, we need to consider the expected CFP time that is required for serving packets that have been generated by the node and neighbours during the time period $\frac{L}{H_x}$.

- **Computing expected number of retransmissions:** The value of σ_x gives the expected number of retransmissions that is required for a successful transmission of a packet generated by node x, which is located at hop h. This value is obtained as [40]

$$\sigma_x = \begin{cases} \frac{1}{1-\epsilon_h}, & \text{for MC traffic,} \\[1.5em] \frac{1}{(1-\chi_x)(1-(1-\alpha_x)^{M+1})}, & \text{for NC traffic.} \end{cases}$$

C. Obtained Reliability over the Path

Based on the derivation of reliability for each hop in (15) and (17), the obtained reliability over each path can be calculated as

$$R_n = \prod_{h=1}^{H_n} R_{r_{hn}}(S). \tag{24}$$

D. Problem Formulation

In order to collect the traffic from SMs either in a single-hop or multi-hop structure, aggregators are placed on top of the existing utility poles. The placement should be conducted such that coverage for all automated devices is ensured, the required latency for critical and non-critical traffic is satisfied, and at the same time, a cost-efficient infrastructure in terms of installation and maintenance is obtained.

To formulate the associated optimization problem let us assume N_{SM} is the number of SMs in the area which need to be covered and N_{poles} is the number of poles from which a subset should be selected for DAP placement. The binary variable x_j indicates whether a DAP is installed on pole j. Also let the binary variables y_{ij}, $q_{ii'}$ and $z_{ii'}$ indicate whether an SM i is directly connected to the DAP located on pole j, whether a node i' is the immediate parent[1] of another node i, and whether node i' is an ancestor of another node i, respectively.

Using these variables and the expressions from Sections II-A to II-C, we can write the optimization problem for the DAP placement in (25) (on the next page). According to [21], [23], DAPs are very costly to be installed. Therefore, in order to have a cost-efficient infrastructure, we define the objective (25a) as the minimization of the installation cost, c_{inst}, which we consider linearly proportional to the total number of DAPs that should be mounted on top of the poles. Assuming that discovering one route is enough for each SM, constraint (25b) ensures that it is

---

[1]Any node which is on the route from the source to the destination is defined as the ancestor of the source. The ancestor node directly connected to the source is called the source's parent.

either directly connected to a DAP or it has an immediate connection to another smart meter, which becomes its parent node. Constraint (25c) provides the relation between the parent of a node, $q_{ii'}$, and its ancestors, $z_{ii'}$. Constraints (25d) and (25e) ensure that only one of the nodes i or i' can be the parent or an ancestor of the other one. Constraint (25f) enforces the connectivity of all nodes to a DAP, via single or multi-hop communication. Accordingly, constraint (25g) as previously obtained in (24), ensures the satisfaction of the reliability constraint as a cumulative effect of packet success ratio and the latency requirement for both MC and NC traffic, where ρ is the specified required reliability in percentage. Constraint (25h) ensures that the aggregated traffic from the connected nodes to each DAP is less than the offered service rate by the DAP, μ. Constraint (25i) ensures that the relation between DAP selection and placement is maintained, i.e., an SM can only be connected to a pole which is selected for DAP installation.

III. DAP PLACEMENT ALGORITHM

The optimization in (25) is an IP problem and directly solving it has an exponential time complexity with regards to the problem size, i.e., number of variables and constraints [41]. Optimization solvers such as CPLEX [18] and GLPK [19] employ the branch and cut method for solving IP problems. However, the complexity of such algorithms is still high and exponential in the worst case scenario. Therefore for large networks, a lower complexity algorithm is desired [14], [15], [42]. In this section, we propose a new heuristic algorithm, which is partly inspired from [10] and [14], and uses a greedy approach for identifying potential locations for relay placement. We later on, through the results presented in Section IV, show that our proposed algorithm can provide a good solution to the DAP placement problem with a relatively low computational complexity.

The proposed DAP placement algorithm consists of two phases. In the first phase, we address the objective (25a) through approximating the minimum required number of aggregators and their initial locations. This is done through selecting poles that cover the largest number of uncovered

$$\min_{\{x_j\},\{y_{ij}\},\{q_{ii'}\},\{z_{ii'}\}} \quad c_{\text{inst}} = \sum_{j=1}^{N_{\text{poles}}} x_j \tag{25a}$$

Subject to

$$\sum_{j=1}^{N_{\text{poles}}} y_{ij} + \sum_{i'=1}^{N_{\text{SM}}} q_{ii'} = 1, \qquad 1 \le i \le N_{\text{SM}}, \tag{25b}$$

$$q_{ii'} \le z_{ii'}, \qquad 1 \le i, i' \le N_{\text{SM}}, \tag{25c}$$

$$q_{ii'} + q_{i'i} \le 1, \qquad 1 \le i, i' \le N_{\text{SM}}, \tag{25d}$$

$$z_{ii'} + z_{i'i} \le 1, \qquad 1 \le i, i' \le N_{\text{SM}}, \tag{25e}$$

$$\sum_{j=1}^{N_{\text{poles}}} y_{ij} + \sum_{j=1}^{N_{\text{poles}}} \sum_{i'=1}^{N_{\text{SM}}} z_{ii'} y_{i'j} = 1, \qquad 1 \le i \le N_{\text{SM}}, \tag{25f}$$

$$R_i \ge \rho, \qquad 1 \le i \le N_{\text{SM}}, \text{ for MC and NC}, \tag{25g}$$

$$\sum_{i=1}^{N_{\text{SM}}} y_{ij} \lambda_i \le \mu, \qquad 1 \le j \le N_{\text{poles}}, \tag{25h}$$

$$y_{ij} \le x_j, \qquad 1 \le i \le N_{\text{SM}}, \quad 1 \le j \le N_{\text{poles}}, \tag{25i}$$

$$x_j, y_{ij}, z_{ii'}, q_{ii'} \in \{0, 1\}, \qquad 1 \le i, i' \le N_{\text{SM}}, \quad 1 \le j \le N_{\text{poles}}, \tag{25j}$$
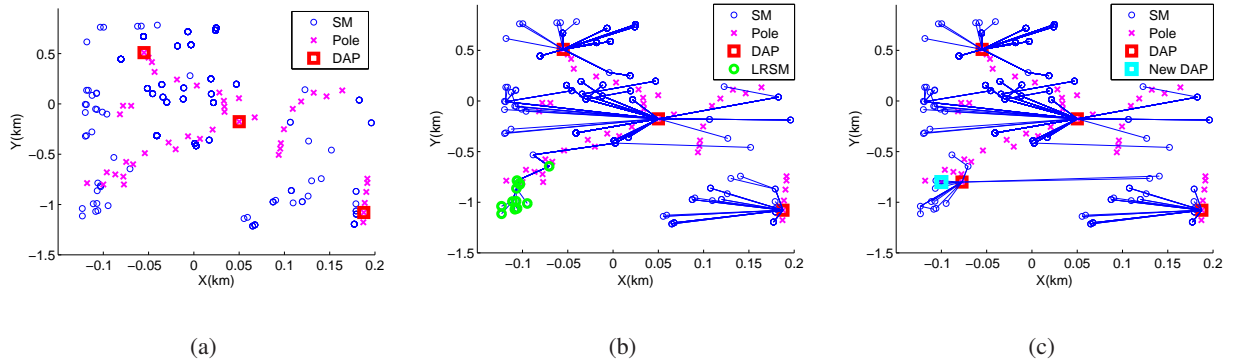

Fig. 2. Sample scenario for illustration of the steps of the heuristic algorithm. (a) First phase pole selection, (b) Second phase - step I, with initial shortest paths (LRSM denotes an SM which experiences low reliability) and second phase - step II (DAP locations have not been changed in this case), (c) Second phase - step III, placing a new DAP at $(-0.1, -0.8)$ and re-running second phase - step I for re-constructing the tree, and second phase - step II, relocating each aggregator closer to the center-point of its current cluster members (the new aggregator is moved to $(-0.07, -0.8)$).

SMs through multi-hop communication as per (25b)-(25f). In the second phase, based on the initial location of DAPs, we explore shortest path routes for the SMs to connect them to the DAPs and ensure that their network coverage, and QoS and capacity requirements as per (25g) and (25h) are maintained.

A. Phase 1: Pole Selection

In this phase, through a greedy approach, we select the poles that have the largest number of connectivities to the uncovered SMs as candidates for DAP installation. In order to identify the set of SMs that can be covered by a certain pole through multi-hop communication as per (25b)-(25f), we construct a k-dimensional (KD) tree[2] over the set of SMs and perform range search operations, considering the effective coverage range of poles and SMs, d_{smax} and d_{pmax}.

We repeat the above step for the remaining SMs that are not yet connected to a selected pole until all SMs are connected to a DAP or there is no solution for the remaining nodes, i.e. there is no pole or SM in their communication range.

B. Phase 2: Tree Construction

In this phase, we connect endpoints to the aggregators that have been selected in phase 1 and ensure that the capacity and QoS requirements (25g) and (25h) are satisfied. We perform the following steps.

Step I (route discovery): We use the Dijkstra algorithm to connect each SM through single or multi-hop communication, to the DAP that its capacity has not yet exceeded as per (25h) and also results in obtaining the maximum packet success rate. To this end, we use the link PERs obtained from (1) and (2) via

$$c_{ij} = \log\left(\frac{1}{1 - \epsilon_{ij}}\right)$$

---

[2]A KD tree is a data structure for organizing k-dimentional data points in a binary search tree [43]. Performing range search operation over this tree (data structure) helps to identify the set of nodes that are in the communication range of certain locations.

as the link costs. This step determines the clusters, i.e., the set of SMs that are connected to each DAP.

Step II (relocating each aggregator to the center-point of its cluster members): As the first phase of the algorithm only addresses the coverage constraint, in this phase we move each DAP to the pole nearest to the center-point of its cluster members, so that on average fewer hops would be required for SMs within the cluster to access the DAP and accordingly, a better reliability can be provided for them. Note that all the SMs should be able to connect to the newly selected location for the DAP, otherwise, this re-location would not be conducted.

Step III (adding new aggregators): In this step, we compute the obtained reliability as per (25g) for all the nodes and disconnect those that experience low reliability for either of their MC or NC traffic. Then, we re-run the first phase of the algorithm for finding new aggregators for covering the disconnected nodes. As there might be some already connected nodes whose reliability would improve if they connected to the newly added aggregators, we repeat the second phase of the algorithm over the whole set of SMs in order to re-connect them to the new set of DAPs. Adding new aggregators can only increase satisfaction of the reliability constraint, and thus this step is re-iterated until the required reliability is met for all nodes or no solution can be found (i.e., no solution exists for meeting the required reliability).

Figure 2 shows an example of the phases of our algorithm in an SGCN with 425 SMs and 45 poles. The smart meters are shown as circles, poles are marked with crosses and the selected DAPs are represented as squares. As it can be seen, the first phase of the algorithm selects three poles for DAP installation (Figure 2(a)). The second phase of the algorithm constructs initial shortest paths for all the nodes and computes their obtained packet success ratio and reliability. We can observe that 13 nodes become disconnected during step III of phase 2 (marked as larger (green) circles in Figure 2(b)) as their obtained reliability with the current set of DAPs is less than the specified reliability of $\rho = 98\%$. Then, through repeating the first phase of the algorithm, a new pole is selected for the DAP placement (new DAP in Figure 2(c)) and steps I and II

of the second phase are repeated for reconstructing the shortest paths and moving poles to the center-point of their currently allocated cluster members.

In this section, we provide details on the performance of the proposed algorithm in terms of optimality and convergence speed.

1) Optimality Analysis: The DAP placement is an instance of the set cover problem [44, Theorem1] and we have applied a greedy approach for solving it. It is well-known that the approximation factor of greedy algorithms for solving a set cover problem in the worst-case scenario is $\ln(N)$, where N is the number of nodes to be covered [44]. Moreover, there is no approximation algorithm that can provide a significantly better approximation factor than what is provided by a greedy algorithm for solving a set cover problem [45]. Therefore, the solution provided by the proposed heuristic algorithm in the worst case differs from the optimal solution by a factor of $\ln(N_{\mathrm{SM}})$, and this is the best approximation factor that a polynomial solution can achieve.

2) Convergence Analysis: According to the global convergence theorem, an algorithm converges to a desired solution if we can define a descent function on the solution set [46]. Since in each iteration of our algorithm, the number of nodes that are not covered by a DAP are decreasing (adding new DAPs improves the experienced reliability), we can conclude that our algorithm converges.

In terms of the convergence ratio, assume r_k is the number of DAPs in the kth iteration of the algorithm, and r^* is the number of DAPs when the algorithm converges. Since in our algorithm, $\nu = \lim_{k\to\infty} \frac{r_{k+1}-r^*}{r_k-r^*}$ is a value between 0 and 1 (as the distance to the required number of DAPs is decreasing), according to [47] we can conclude that the algorithm linearly converges to the desired solution with ratio ν. The value of ν is different for different scenarios. For a smaller value of ν, the algorithm converges faster.

TABLE II

SIMULATION PARAMETERS [4], [27].

| Parameter	Value	Parameter	Value	Parameter	Value
Req. reliability, ρ	90%	PL model	Erceg Type B	Bandwidth	281 kHz (802.15.4g)
N_{ARQ}	4	Interference Margin (I_{m})	6 dB	Fading Margin (η)	12.3 dB
SM / DAP height	2 / 10 m	Transmission power (P_{tx})	30 mW	Modulation and	QPSK
Noise Factor (F)	7 dB	Receiver Noise PSD (N_0)	−174 dBm/Hz	coding scheme (MCS)	code rate of $\frac{3}{4}$

IV. NUMERICAL RESULTS AND DISCUSSION

In this section, we test our proposed DAP placement algorithm using realistic smart meter and pole locations information from the area of Kamloops, BC, Canada.

A. Simulation Settings

Table II summarizes the parameters we have used for running our simulations. Figure 3(a) presents the geographical locations of SMs and poles over the map of Kamloops, BC, Canada. The SMs and poles are marked with blue circles and magenta crosses, respectively. It is important to note that the poles are mostly aligned with the roads on the map and their location do not follow a uniform-random distribution model that is sometimes assumed in the literature. As suggested in [4], the Erceg Type B best models the signal propagation for the smart grid infrastructure in rural and suburban areas. Therefore, we have used this model for emulating the pathloss in the considered Kamloops suburban area, which is a hilly environment with light to moderate number of trees. The area size is 20×2 km^2 which includes 8053 SMs and 776 poles. The traffic specifications are derived from [3] as presented in Table I in Section II.

B. Performance Comparison with CPLEX

We first compare the optimality and complexity of our devised algorithm with the results obtained based on the CPLEX software for solving (25). To this end, since CPLEX is not able to solve the large-scale scenarios, we select smaller scale scenarios considering different area densities from the Kamloops scenario. The performances of our algorithm and the CPLEX
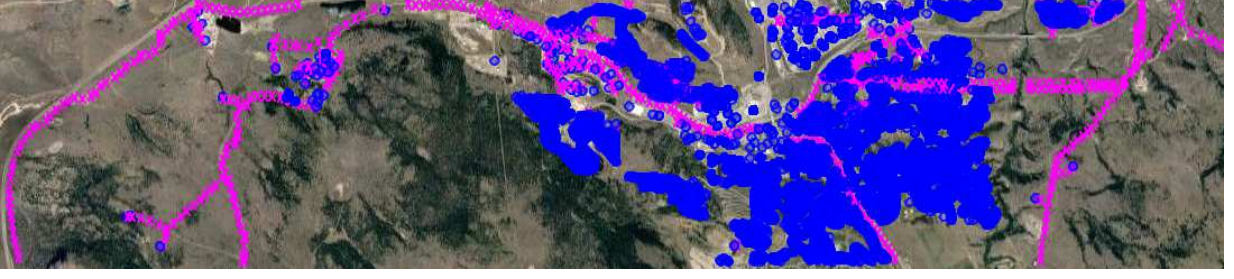
(a)
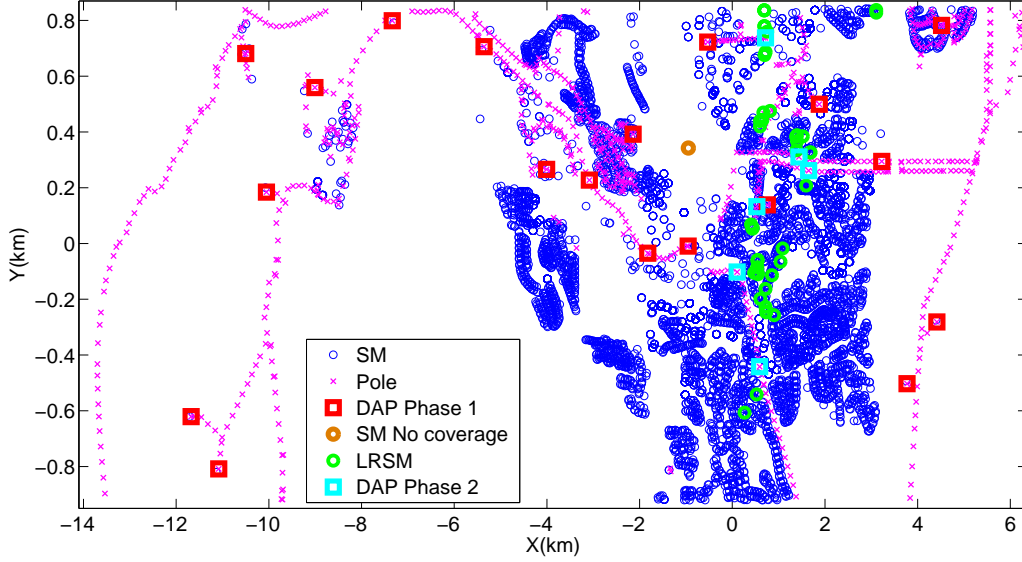
(b)

Fig. 3. (a) The geographic location of smart meters and poles in the Kamloops suburban area. (b) Results of the proposed DAP placement algorithm for the Kamloops scenario. The red and cyan squares show the poles that are selected for DAP placement respectively in the first and second phase of the algorithm. The green circles show the low-reliability SMs (LRSMs) for which the poles in the second phase were added. The larger (orange) circle identifies the 19 SMs that are not connected to any DAP.

software are compared in Table III. As the number of aggregators indicates the optimization objective, we can observe that our algorithm returns near-optimal results and at the same time, our algorithm offers much lower run-time complexity and memory requirement. We further observe from Table III that more aggregators are required for the scenarios with lower SM density.

TABLE III

Comparing the optimality and complexity of proposed DAP placement algorithm and CPLEX for solving

problem (25)

Scenario	Method	Memory (MB)	Time (sec.)	Number of Iterations	Number of Aggregators	Max. hops
47 SMs 43 Poles Rural (23.5 SMs per km^2)	CPLEX	358.2 4487 Variables (13009 Non-zero coeffs.) 6746 Constraints	25.0	NA	4	2
47 SMs 43 Poles Rural (23.5 SMs per km^2)	DAP placement algorithm	0.7	5.0 4.3 First phase 0.7 Second phase	1	4	2
60 SMs 12 Poles Suburban (155.2 SMs per km^2)	CPLEX	481.1 4124 Variables (12942 Non-zero coeffs.) 7841 Constraints	77.0	NA	1	10
60 SMs 12 Poles Suburban (155.2 SMs per km^2)	DAP placement algorithm	0.9	6.1 5.1 First phase 1.0 Second phase	2	2	6
74 SMs 37 Poles Suburban (513.9 SMs per km^2)	CPLEX	1094.6 9554 Variables (34290 Non-zero coeffs.) 15140 Constraints	1860.0 (Stopped at 6% optimality gap)	NA	1	5
74 SMs 37 Poles Suburban (513.9 SMs per km^2)	DAP placement algorithm	1.2	7.3 6.5 First phase 0.8 Second phase	1	1	5
161 SMs 24 Poles Urban (958.3 SMs per km^2)	CPLEX	854.3 38117 Variables (135888 Non-zero coeffs.) 64335 Constraints	840.0	NA	1	6
161 SMs 24 Poles Urban (958.3 SMs per km^2)	DAP placement algorithm	2.3	14.3 10.3 First phase 4.0 Second phase	1	1	6

C. Validation of the Delay Model

In order to validate our assumptions and delay model derived in Section II, we use the network simulator-3 (NS3) [48] software to simulate the SM-to-relay transmissions in the Kamloops scenario. Each SM generates packets based on the traffic classes listed in Table I. We measure the total delay experienced by each packet as the difference between the time it is successfully received by the destination and its generation time. Figure 4 compares the empirical delay distribution with the analytical probability of delay satisfaction for the packets that have been generated from an SM, which has 124 feeding nodes and 126 neighbour nodes. Nine of the

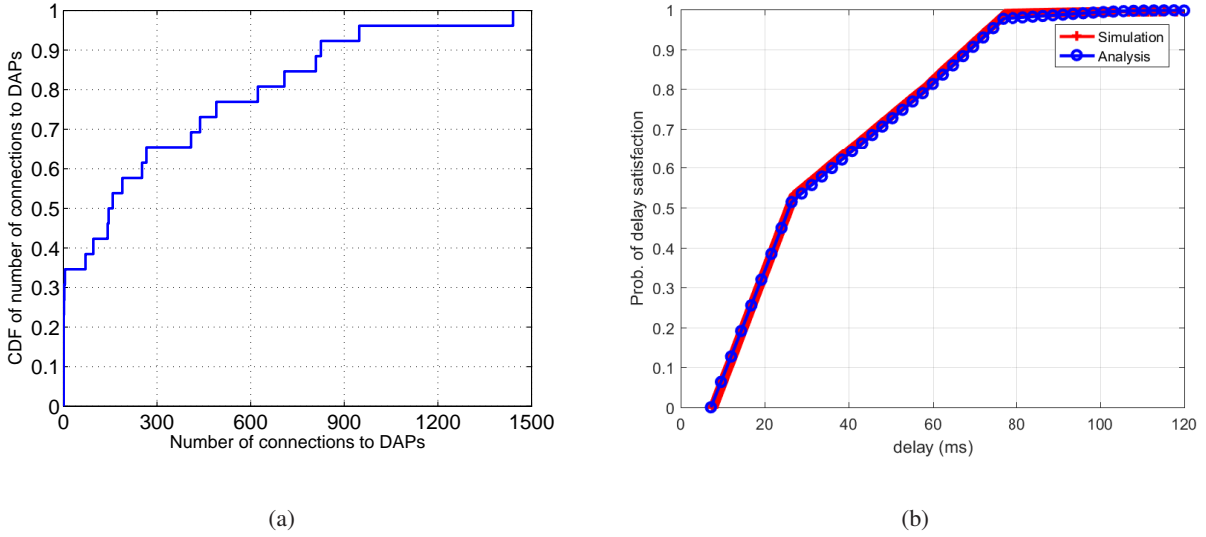(a)   (b)

Fig. 4.   (a) CDF of the total number of connections to DAPs (the mean value is 322). (b) Comparison of the analysis and simulation for the probability of delay satisfaction as a function of deadline.

neighbours have respectively 1244, 330, 319, 233, 108, 58, 53, 26, 5 feeding nodes and the other 117 nodes do not have any. As it can be seen from Figure 4, the probability of latency satisfaction obtained from simulations closely matches the values obtained from the analysis in Section II-B. This verifies that the assumptions made in the system model are valid for the traffic classes listed in Table I. Specifically, under the mixed traffic model the distribution of packet generations in each SM can be well approximated with a Poisson distribution, and the distribution of packet arrival in the forwarding nodes can be also assumed to follow a Poission distribution.

D. Number of DAPs

Figure 3(b) shows the result of the DAP placement algorithm for the whole Kamloops scenario. In the first iteration of the algorithm, 19 poles, marked with red squares, are selected for DAP placement such that the network coverage can be ensured. In the next 3 iterations, 6 additional poles, marked with cyan squares, are added in order to enforce the required reliability for the SMs that do not satisfy the reliability requirement. These SMs are marked with green circles
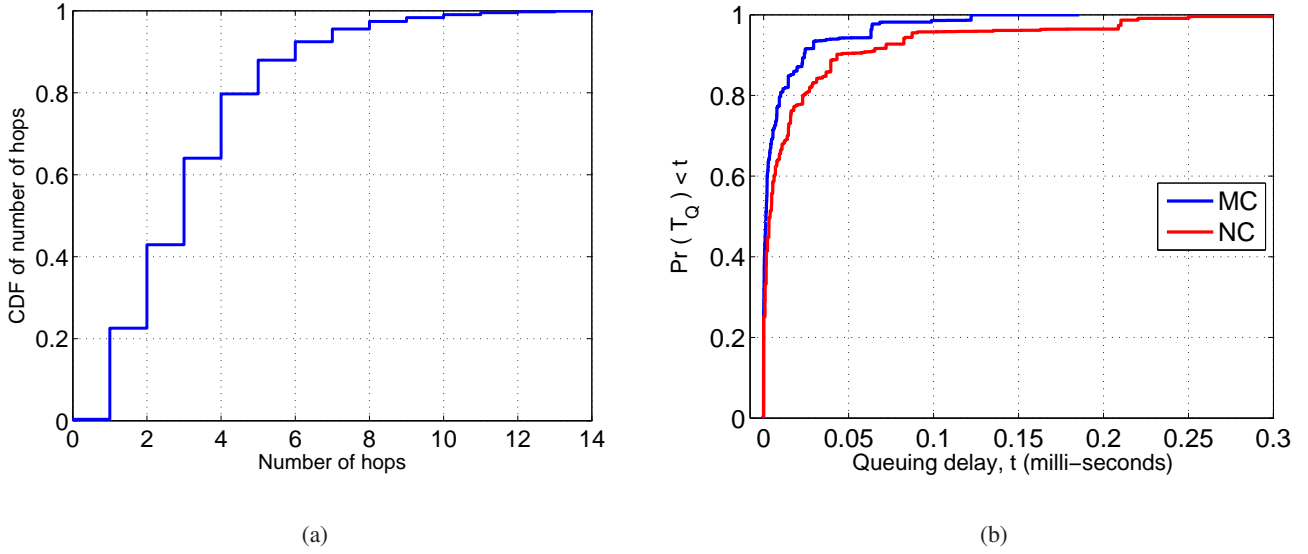
(a)   (b)

Fig. 5.   (a) CDF of the number of hops (the mean value is 3.2). (b) CDF of queuing delay for the MC and NC traffic.

in the Figure. For the 19 SMs which are located in the same building at location $(-1.0, 0.35)$, there is no connectivity solution, as there is no pole or SM in their connectivity range.

E. Connections per Pole

Figure 4(a) shows the empirical cumulative distribution function (CDF) of the number of connections to the DAPs for the Kamloops scenario. It is observed that around 80% of the DAPs have less than 623 SM connections. We also note that about 35% of the DAPs have less than 5 connections which is due to the several rural areas with sparse location of smart meters, e.g. for $x < -6.0$ in Figure 3. To reduce the number of DAPs with few connectivities, the installation of range extenders would be beneficial.

F. Number of Hops

Figure 5(a) shows the distribution of the number of hops for SM-DAP connections in the network for the Kamloops scenario. As can be seen, around 22% of the nodes are directly connected to DAPs, and 90% of the nodes are within a 6-hop connectivity from a DAP. For the farther nodes, our algorithm ensures that their obtained reliability is still within what is required.

This shows the flexibility of our algorithm compared to [10] and [20], where they address latency through considering a fixed number of hops, while our algorithm selects the DAP locations and number of hops based on the network topology, SM to SM and SM to pole distances and number of competitors at each hop. The dynamic selection of number of hops based on these parameters makes it possible to access farther SMs with the lowest number of DAPs, without compromising the required latency.

G. Queuing Delay

Figure 5(b) shows the CDF of the queuing delays observed for the mission-critical and non-critical traffic for the Kamloops scenario. The maximum queuing delay observed for mission-critical traffic is around 0.17 ms and the maximum queuing delay observed for non-critical traffic is around 0.30 ms. The small queuing delay observed is due to the low data rate at the nodes.

H. Complexity Analysis

1) Proposed algorithm: Here we estimate the complexity of each step in our algorithm to derive its overall complexity.

KD tree construction and range search: In the first phase of our placement algorithm, we use the KD tree data structure for storing SM locations. Then, we perform a range search operation over this tree in order to identify the set of SMs which are in the communication range of a certain pole. The runtime and memory complexity of KD tree construction are respectively $O(N_{\mathrm{SM}} \log(N_{\mathrm{SM}}))$ and $O(N_{\mathrm{SM}})$. The range search operation complexity is $O(N_{\mathrm{poles}} \log(N_{\mathrm{SM}}))$.

Shortest path: In order to identify optimal routes for each SM, shortest paths are constructed from each DAP using the Dijkstra algorithm. The associated time and memory complexity are $O(N_{\mathrm{SM}}^2)$ and $O(N_{\mathrm{SM}}^2)$, respectively.

Since the shortest-path search has the higher complexity of the above two steps, the total algorithm run-time and memory complexities are of the orders of $N_{\mathrm{DAP}} O(N_{\mathrm{SM}}^2)$ and $N_{\mathrm{DAP}} O(N_{\mathrm{SM}}^2)$, respectively. For the specific Kamloops scenario with 8053 SMs considered above, we measured a memory usage of 83 MB.

2) CPLEX: CPLEX uses a branch and cut algorithm for finding the optimal solution to the IP problem. In the worst case, the complexity of such an algorithm is exponential, and the actual mean time-complexity depends on many factors and is evaluated empirically [49], [50]. Another limiting factor when optimization solvers are used for solving IP problems is the required RAM. According to [51], for every 1000 constraints, at least 1 MB RAM is required by CPLEX in order to solve an IP problem. Since the presented DAP problem in (25) considering 8053 SMs and 776 poles has around 140,000,000 constraints, an estimated 140 GB RAM would be needed to solve it by CPLEX.

I. Comparison with Other Works

In this section, we compare the optimality and time-complexity of our algorithm with the work presented in [21] and [25]. For a fair comparison with [21], we limit the number of hops to $H = 4$ and compare the solution of our algorithm with the second scenario in [21, Table II] that has a similar number of SMs and poles as the Kamloops scenario. We observe from our simulations results, which are omitted here due to space constraints, that our algorithm finds a more cost-efficient solution as it only selects 37 out of 776 poles and ensures coverage and latency constraints, while the algorithm from [21] selects 426 poles for DAP placement and only ensures SM coverage. Furthermore, the complexity of their algorithm is higher. In particular, the method from [21] requires to calculate the multi-hop connectivity matrix as part of the pre-processing method, which has a computational complexity of $H \cdot O((N_{\text{poles}} + N_{\text{SM}})^3)$. Then, the coverage matrix is passed to the GLPK software for obtaining the minimum number of cover sets, which in the worst-case scenario has a complexity of $O(2^{(N_{\text{poles}}+N_{\text{SM}})})$. When the network becomes large, their heuristic algorithm breaks the area into smaller squares which can be handled by the optimizer. Their post-optimization step involves merging the solution of smaller squares, solved by GLPK software, by removing the redundant poles located in square edges. This step has the complexity of $O(N_{\text{SM}}N_{\text{poles}}^2)$. In terms of the memory complexity, the

method from [21] would require 2-306 MB depending on the selected square size.

Reference [25] utilizes the divide and conquer algorithm for identifying the set of SMs that can relay traffic in an AMI. In the procedure of relay selection, the maximization of QoS is considered in the objective by minimizing packet loss and average latency, which are calculated based on the link distance and M/D/1/k queueing theory. The algorithm focuses on single-hop connectivity of endpoints to the aggregator and finally, connects every 10-15 endpoints to one aggregator. This is not a feasible solution in practice, since at least around 533 aggregators would then need to be installed and maintained.

V. Conclusion

In this paper, the problem of DAP placement for an AMI with overhead power lines has been investigated. We proposed a mutli-phase heuristic algorithm for selecting the optimized pole locations for DAP placement such that smart grid QoS requirements can be met. We maximize the obtained reliability for the smart grid traffic through discovering routes with minimum packet error rates and scheduling the mission-critical and the non-critical traffic using TDMA and CSMA/CA protocols, respectively. The probability of exceeding a certain latency is computed based on the specific characteristics of these two protocols. Comparing the results of our algorithm with the literature and solutions obtained by the IBM CPLEX software for small-scale examples, we believe that our algorithm is competitive in terms of performance for the problem at hand, albeit at much lower complexity. The complexity advantage allows us to successfully tackle larger-scale problems as shown in this paper.

VI. Acknowledgment

REFERENCES

[1] S. S. S. R. Depuru, L. Wang, and V. Devabhaktuni, "Smart meters for power grid: Challenges, issues, advantages and status," *Renewable and Sustainable Energy Reviews*, vol. 15, no. 6, pp. 2736–2742, 2011.

[2] S. Darby, "The effectiveness of feedback on energy consumption," *A Review for DEFRA of the Literature on Metering, Billing and direct Displays*, vol. 486, 2006.

[3] "SG Network System Requirements Specification v5.1," Open Smart Grid (Open SG), Tech. Rep., 2010. [Online]. Available: http://osgug.ucaiug.org/default.aspx.

[4] "NIST PAP2 guidelines for assessing wireless standards for smart grid application," 2012. [Online]. Available: http://tinyurl.com/NIST-PAP2-Guidelines

[5] S. McLaughlin, D. Podkuiko, and P. McDaniel, "Energy theft in the advanced metering infrastructure," in *Critical Information Infrastructures Security*. Springer, 2009, pp. 176–187.

[6] WiMAX Forum, "WiMAX forum system profile requirements for smart grid applications: Requirements for WiGRID," 2013. [Online]. Available: http://tinyurl.com/Smart-Grid-Utility-Requirement

[7] G. Atkinson and M. Thottan, "Leveraging advanced metering infrastructure for distribution grid asset management," in *IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2014, pp. 670–675.

[8] A. Bartoli, J. Hernández-Serrano, M. Soriano, M. Dohler, A. Kountouris, and D. Barthel, "Secure lossless aggregation for smart grid M2M networks," in *IEEE International Conference on Smart Grid Communications*, 2010, pp. 333–338.

[9] D. Niyato and P. Wang, "Cooperative transmission for meter data collection in smart grid," *IEEE Commun. Mag.*, vol. 50, no. 4, pp. 90–97, 2012.

[10] B. Aoun, R. Boutaba, Y. Iraqi, and G. Kenward, "Gateway placement optimization in wireless mesh networks with qos constraints," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 11, pp. 2127–2136, 2006.

[11] I. Chatzigiannakis, A. Kinalis, and S. Nikoletseas, "Sink mobility protocols for data collection in wireless sensor networks," in *ACM International Workshop on Mobility Management and Wireless Access*, 2006, pp. 52–59.

[12] B. Sheng, Q. Li, and W. Mao, "Data storage placement in sensor networks," in *ACM International Symposium on Mobile Ad Hoc Networking and Computing*, 2006, pp. 344–355.

[13] W. Alsalih, H. Hassanein, and S. Akl, "Placement of multiple mobile data collectors in wireless sensor networks," *Ad Hoc Networks*, vol. 8, no. 4, pp. 378–390, 2010.

[14] B. Lin, P.-H. Ho, L.-L. Xie, X. Shen, and J. Tapolcai, "Optimal relay station placement in broadband wireless access networks," *IEEE Trans. Mobile Comput.*, vol. 9, no. 2, pp. 259–269, 2010.

[15] S. Lee and M. F. Younis, "EQAR: Effective QoS-aware relay node placement algorithm for connecting disjoint wireless sensor subnetworks," *IEEE Trans. Comput.*, vol. 60, no. 12, pp. 1772–1787, 2011.

[16] S. Lee and M. Younis, "QoS-aware relay node placement in a segmented wireless sensor network," in *IEEE International Conference on Communications (ICC)*, 2009, pp. 1–5.

[17] "Communications requirements of smart grid technologies," US Department of Energy, Tech. Rep., 2010.

[18] "IBM CPLEX Optimization Studio," *URL https://tinyurl.com/ibm-cplex-optimization-studio*, 2012.

[19] A. Makhorin, "GLPK (GNU linear programming kit)," 2016. [Online]. Available: http://www.gnu.org/software/glpk/

[20] G. Souza, F. Vieira, C. Lima, G. Junior, M. Castro, and S. Araujo, "Optimal positioning of GPRS concentrators for minimizing node hops in smart grids considering routing in mesh networks," in *IEEE PES Conference On Innovative Smart Grid Technologies Latin America (ISGT LA)*, 2013, pp. 1–7.

[21] G. Rolim, D. Passos, I. Moraes, and C. Albuquerque, "Modelling the data aggregator positioning problem in smart grids," in *IEEE International Conference on Computer and Information Technology (CIT)*, 2015, pp. 632–639.

[22] K. Ali, W. Alsalih, and H. Hassanein, "Set-cover approximation algorithms for load-aware readers placement in RFID networks," in *IEEE International Conference on Communications (ICC)*, 2011, pp. 1–6.

[23] F. Aalamifar, G. N. Shirazi, M. Noori, and L. Lampe, "Cost-efficient data aggregation point placement for advanced metering infrastructure," in *IEEE International Conference on Smart Grid Communications*, 2014, pp. 344–349.

[24] P. Y. Kong, "Wireless neighborhood area networks with QoS support for demand response in smart grid," *IEEE Trans. Smart Grid*, vol. 7, no. 4, pp. 1913–1923, July 2016.

[25] D. Niyato, L. Xiao, and P. Wang, "Machine-to-machine communications for home energy management system in smart grid," *IEEE Commun. Mag.*, vol. 49, no. 4, pp. 53–59, 2011.

[26] "IEEE Standard for Local and metropolitan area networks–Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs) Amendment 3: Physical Layer (PHY) Specifications for Low-Data-Rate, Wireless, Smart Metering Utility Networks," *IEEE Std 802.15.4g-2012 (Amendment to IEEE Std 802.15.4-2011)*, pp. 1–252, April 2012.

[27] "Open Smart Grid (open SG)." [Online]. Available: http://osgug.ucaiug.org/default.aspx.

[28] R. R. Mohassel, A. Fung, F. Mohammadi, and K. Raahemifar, "A survey on advanced metering infrastructure," *International Journal of Electrical Power & Energy Systems*, vol. 63, pp. 473–484, 2014.

[29] F. Gomez-Cuba, R. Asorey-Cacheda, and F. Gonzalez-Castano, "Smart grid last-mile communications model and its application to the study of leased broadband wired-access," *IEEE Trans. Smart Grid*, vol. 4, no. 1, pp. 5–12, Mar 2013.

[30] Participants of the Priority Action Plan 2 working group, "NIST smart grid interoperability panel PAP2 guidelines for assessing wireless standards for smart grid application," 2014. [Online]. Available: http://nvlpubs.nist.gov/nistpubs/ir/2014/NIST.IR.7761r1.pdf

[31] Z. Shi, C. Beard, and K. Mitchell, "Analytical models for understanding space, backoff, and flow correlation in CSMA wireless networks," *Wireless networks*, vol. 19, no. 3, pp. 393–409, 2013.

[32] P. Di Marco, P. Park, C. Fischione, and K. H. Johansson, "Analytical modeling of multi-hop IEEE 802.15.4 networks," *IEEE Trans. Veh. Technol.*, vol. 61, no. 7, pp. 3191–3208, 2012.

[33] S. Ray, D. Starobinski, and J. B. Carruthers, "Performance of wireless networks with hidden nodes: A queuing-theoretic analysis," *Computer Communications*, vol. 28, no. 10, pp. 1179–1192, 2005.

[34] J. Shi, T. Salonidis, and E. W. Knightly, "Starvation mitigation through multi-channel coordination in CSMA multi-hop wireless networks," in *ACM International Symposium on Mobile Ad Hoc Networking and Computing*, 2006, pp. 214–225.

[35] S. K. Bose, *An introduction to queueing systems*. Springer Science & Business Media, 2013.

[36] J. Gao, J. Hu, and G. Min, "A new analytical model for slotted IEEE 802.15. 4 medium access control protocol in sensor networks," in *International Conference on Communications and Mobile Computing*, vol. 2. IEEE, 2009, pp. 427–431.

[37] J. Misic, S. Shafi, and V. B. Misic, "Performance of a beacon enabled IEEE 802.15.4 cluster with downlink and uplink traffic," *IEEE Trans. Parallel Distrib. Syst.*, vol. 17, no. 4, pp. 361–376, 2006.

[38] M. Baz, P. D. Mitchell, and D. A. Pearce, "Analysis of queuing delay and medium access distribution over wireless multihop PANs," *IEEE Trans. Veh. Technol.*, vol. 64, no. 7, pp. 2972–2990, 2015.

[39] M. Fernandez and S. Williams, "Closed-form expression for the Poisson-Binomial probability density function," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 46, no. 2, pp. 803–817, April 2010.

[40] D. Bertsekas and R. Gallager, *Data Networks (2Nd Ed.)*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1992.

[41] W. Meng, R. Ma, and H.-H. Chen, "Smart grid neighborhood area networks: a survey," *IEEE Netw.*, vol. 28, no. 1, pp. 24–32, 2014.

[42] W. Alsalih, S. Akl, and H. Hassanein, "Placement of multiple mobile data collectors in underwater acoustic sensor networks," in *IEEE International Conference on Communications (ICC)*, 2008, pp. 2113–2118.

[43] H. Samet, *The design and analysis of spatial data structures*. Addison-Wesley Reading, MA, 1990, vol. 199.

[44] R. Chandra, L. Qiu, K. Jain, and M. Mahdian, "Optimizing the placement of internet taps in wireless neighborhood networks," in *IEEE International Conference on Network Protocols*, 2004, pp. 271–282.

[45] U. Feige, "A Threshold of Ln N for Approximating Set Cover," *Journal of the ACM*, vol. 45, no. 4, pp. 634–652, Jul. 1998.

[46] W. I. Zangwill, *Nonlinear programming: A unified approach*. Prentice-Hall, 1969.

[47] D. G. Luenberger, *Introduction to linear and nonlinear programming*. Addison-Wesley Reading, MA, 1973, vol. 28.

[48] "Network Simulator-3 (NS-3)," 2015. [Online]. Available: http://www.nsnam.org/

[49] "Complexity of the integer linear programming," 2014. [Online]. Available: https://www.ibm.com/developerworks/community/forums/html/topic?id=77777777-0000-0000-0000-000014393637

[50] "How to measure the difficulty of a mixed-linear integer programming (MILP) problem," 2015. [Online]. Available: https://www.researchgate.net/post/How_to_measure_the_difficulty_of_a_Mixed-Linear_Integer_Programming_MILP_problem

[51] "Guidelines for estimating CPLEX memory requirements based on problem size," 2012. [Online]. Available: http://www-01.ibm.com/support/docview.wss?uid=swg21399933