Scalable Meter Data Collection in Smart Grids Through Message Concatenation

Babak Karimi, *Student Member, IEEE*, Vinod Namboodiri, *Member, IEEE*, and Murtuza Jadliwala, *Member, IEEE*

***Abstract*—Advanced metering infrastructure (AMI) initiatives are a popular tool to incorporate changes for modernizing the electricity grid, reduce peak loads, and meet energy efficiency targets. There is the looming issue of how to communicate and handle consumer data collected by electric utilities and manage limited communication network resources. Several data relay points are required to collect data distributedly and send them through a communication backhaul. This paper studies the smart meter message concatenation (SMMC) problem of how to efficiently concatenate multiple small smart metering messages arriving at data concentrator units in order to reduce protocol overhead and thus network utilization. This problem needs to deal with the added constraint that each originating message from its source may have its own stated deadline that must be taken into account during the concatenation process. This paper provides hardness results for the SMMC problem, and proposes six heuristics and evaluates them to gain a better understanding of the best data volume reduction policies that can be applied at data concentrators of AMI infrastructures. These results are further tested for feasibility under practical settings based on aspects, such as network and processing delays, tightness of application deadlines, and lossy backhaul links.**

***Index Terms*—Advanced metering infrastructure (AMI), algorithms, communication networks, data management.**

I. Introduction

THE INFORMATION communication and control layer of the smart grid brings about numerous advances, including the empowerment of customers to actively participate in the maintenance of the supply-demand balance around the clock and the resulting reliability improvement in electricity service. There are many benefits to grid operators, consumers, and society as a whole from adopting advanced metering infrastructure (AMI) technologies [1]. With the introduction of AMI technology, two-way communication between a "smart" meter and the grid operator's control center, as well as between the smart meter and consumer appliances, would be facilitated for various applications [2]. Besides AMI, there are many other applications that will be enabled by information flow across the electric power grid. These include distributed generation, state estimation of the power distribution system, and demand-side management, to name a few.

A big challenge for smart grid application scenarios, and the information-sharing framework that enables them, will be handling the massive amount of data that is expected to be collected from data generators and sent through the communication backhaul to the grid operator. For example, by current standards, each smart meter sends a few kilobytes of data every 15–60 min to grid operators [3], [4]. When this is scaled up to many thousands, existing communication architectures will find it difficult to handle the data traffic due to the limited network capacities, especially in limited bandwidth last mile networks [5], [6]. Future applications may require data to be collected at a finer granularity, thus adding to the challenge [7]. Network capacity is a precious resource for electric utilities because they are either leasing such networks from third-party providers [8], or building infrastructure themselves and leasing bandwidth out (especially at the backhaul) to recuperate investment costs [9]. In either case, it is in the interest of electric utilities to reduce the volume of information transported through these networks for smart grid applications while ensuring quality-of-service (QoS) requirements are met.

One approach to reduce data volume given some application sampling rate is to concatenate multiple messages into a larger packet to reduce protocol overhead due to packet headers. This approach has the potential to reduce network capacity requirements significantly (quantified later in this paper) due to the small size of messages sent in smart metering networks, with packet headers possibly being of a comparable size to the underlying message to be sent. Such concatenation of messages can be done by each smart meter itself. However, each meter may not generate messages frequently enough to be able to have the chance to concatenate enough packets to reduce overheads significantly and also meet their stated application deadlines. Each meter is also expected to be relatively constrained (compared to a concentrator) in terms of data storage capabilities to keep a large window of packets from which to aggregate. Thus, a better approach is to concatenate messages at an intermediate point upstream from individual meters.

Such an intermediate point where message concatenation can be done is at data concentrator units (DCUs) (or some similar entity, sometimes also called a data aggregator) that collect data from many smart meters and forward them upstream. Fig. 1 depicts this concept and shows the DCUs role at the power-distribution level of the power grid. Data concentrators

Manuscript received February 28, 2014; revised September 10, 2014 and January 29, 2015; accepted March 30, 2015. Date of publication May 13, 2015; date of current version June 18, 2015. This work was supported by the Power Systems Engineering Research Center under Project S-54. Paper no. TSG-00173-2014.

The authors are with the Department of Electrical Engineering and Computer Science, Wichita State University, Wichita, KS 67260 USA (e-mail: bxkarimi@wichita.edu; vinod.namboodiri@wichita.edu; murtuza.jadliwala@wichita.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSG.2015.2426020

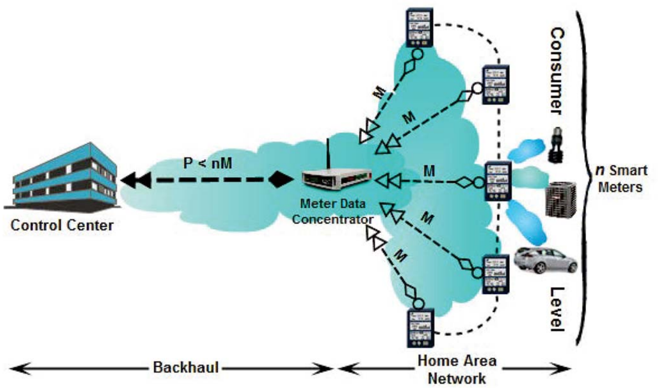

Fig. 1. DCUs envisioned role of message concatenation at the power distribution level.

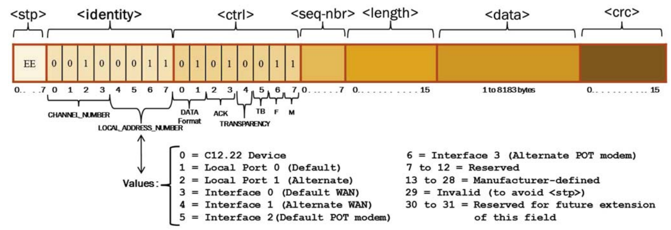

Fig. 2. Smart meter datagram structure.

or aggregators can play an important role in reducing network capacity requirements by reducing packet protocol overhead through message concatenation algorithms applied along the data collection tree. Such algorithms and policies, however, do not exist currently and need to be developed keeping in mind the unique characteristics of metering data like variable packet sizes, stochastic arrivals, and the presence of messages with and without deadlines. Current DCUs on the market lack the ability to reduce the volume of data flowing through them and real-time aggregation capabilities. They only provide simple integration of sensing and wide area network communications options with the intention to follow the PRIME standard [10] which gives the utilities the freedom to choose meters from various vendors and avoid being reliant on proprietary solutions from a single source.

In this paper, we design and comparatively evaluate a suite of online message concatenation algorithms at DCUs in the AMI scenario that minimize usage of network capacity in transporting data through the meter data collection network while meeting QoS constraints imposed by applications on individual messages. The specific contributions of this paper include the following:

1) formulation of the message concatenation problem at DCUs in smart metering networks to minimize network capacity utilization;
2) hardness results for the formulated message concatenation problem that proves it as NP-complete;
3) six different heuristic-based algorithms that can be employed at DCUs for the message concatenation problem;
4) comparative performance evaluation of proposed heuristic-based algorithms for message concatenation;
5) exploration of feasibility of message concatenation under practical settings considering network and processing delays, tighter application deadlines, and lossy backhaul links.

Our results indicate that the proposed heuristic-based concatenation algorithms can reduce data volume in the range of 10%–25% for typical backhaul technologies used, with greater benefits seen for scenarios with higher data traffic rates. These benefits are obtained operating only on packet headers without compressing or aggregating the underlying information in messages. Our results are also shown to hold up well under various practical issues such as network and processing delays, tighter application deadlines, and lossy backhaul links.

II. Problem Formulation

A. Motivation

In most communication protocol suites (e.g., TCP/IP) used for sending smart metering messages, the small size of packets will result in a high amount of protocol overhead due to packet headers. For example, for messages of size 100 bytes from the source smart meter, there may be 40–60 bytes of additional header overheads due to TCP/IP protocols and specific versions used. If a data concentrator collects multiple packets and strips off all individual headers and includes only one header for the larger aggregated message, there could be significant reductions in network capacity utilization. Studying the messaging format for the ANSI C12 smart meter communications standard in [11] provides an idea of message sizes involved and the amount of protocol overhead to expect. As shown in Fig. 2, each smart meter generated message includes parameters like meter identification number, equipment status, type of message, among others. This information is enough to uniquely identify a message source with no additional protocol header information required for source identification. Thus, source protocol headers can be stripped away to rely only on a common aggregated packet header to route the packet to the destination.

In Table I (abstracted from [5]), basic message types along with their properties are listed. It can be seen that messages can be of various sizes (from 20–500 bytes), and can have loose or strict deadlines (2–5 s), or no deadlines at all. Some messages may be generated randomly at any time to indicate critical events that need to be responded to immediately. Data concentrators will have the challenge of handling these varying message sizes that may or may not have deadlines, with possibly stochastic arrivals, at the same time guaranteeing that each message meet any specified deadline. Stochastic message generation and critical events with short deadlines exclude the use of polling based algorithms to collect data at DCUs.

B. Related Work

There have been much prior work on data aggregation in the field of wireless sensor networks (WSNs) [12]. Typical approaches to WSNs have focused on efficient data gathering and energy-latency tradeoffs under deadline

TABLE I
SMART METER DATA MESSAGE TYPES

Message/ Traffic Description	Size (Bytes)	Inter-arrival interval	Inter-arrival unit	Delay Objective
Meter clock sync.	64	Day	1	2 secs
Interval data read	480	Day	1	Best effort
Firmware patch/ upgrade confirmation/ acknowledge	20	Year	1	Best effort
Meter ping (on demand read)	64	Week	4	2 secs
Meter remote diagnostic	500	Day	4	2 secs
Tamper notification	64	Week	26	5 secs
Meter remote disconnect/ reconnect response	500	Day	1	2 secs

constraints (see [13]–[15]). These schemes propose algorithms for grouping smaller packets into larger ones by delaying data transmissions at the relaying nodes whenever slack times are positive with significant reductions in packet transmissions, congestion, and battery energy use. In this paper, our goal is similar in proposing data concentration at DCUs as relay nodes. However, power or energy consumption of the nodes employed are not considered because the AMI infrastructure is expected to have access to electric power at all times with backup batteries. This shifts the focus of the problem from battery life of nodes involved to the reduction of network capacity utilization. Reference [6] does look at data volume reduction in smart metering networks, but does not include aspects such as message concatenation.

C. Smart Metering Message-Concatenation Problem

The smart metering message concatenation (SMMC) problem considered in this paper is as follows. A DCU receives different types of messages from smart meters with a stochastic arrival process (we will discuss this arrival process later in Section IV). Each message can be of a different size and comes with an application specific end-to-end deadline by which it must reach the common destination that is the utility control center. Each message has protocol overhead as it is packaged into a packet before being sent to the DCU. The DCU can either send each packet to the destination as it arrives as a single message or wait and concatenate multiple messages before sending them out over the backhaul to the destination. The objective considered is to minimize the number of individual packets (and hence protocol overhead) sent upstream by the DCU so as to reduce network capacity requirements of the backhaul. The constraints are that all packets meet their deadline (if any) and that each concatenated packet generated (including a common packet header) has a upper size limit, W, governed by the maximum transmission unit (MTU) of the upstream link from the DCU. The objective function chosen helps reduce total overhead required to send all messages within a given time period T by maximizing the size of each concatenated packet for a fixed header size H. In this paper, we assume that messages are not compressed from their original sizes (zero-compression) and the solution to the SMMC problem at DCUs would serve as a lower bound for the possible reduction in network utilization by additional schemes (possibly that compress message sizes themselves) developed in the future for the smart metering scenario. We focus on only a single DCU and its concatenation operation in this paper; in future work, we envision considering a more wider view of the backhaul network and the use of multilevel DCUs along the communications network.

A formal statement of the SMMC problem is provided in the following definition.

Definition 1: Assume that over some period of time T, all smart meters together generate n messages $M = \{m_1, \ldots, m_n\}$. Each message $m_i \in M$ has size s_i and an associated protocol header h_i accompanying it till the DCU with (s_i, h_i, $s_i + h_i \in [0, W]$), an arrival time at the DCU of a_i ($a_i \in [0, T]$), and a deadline d_i ($d_i \in [a_i, \infty]$) by which it must leave the DCU, where $i = 1 \cdots n$. Then, the SMMC problem is to determine an integer number of packets $k (k \leq n)$ and a k-partition $P_1 \cup P_2 \cup \cdots \cup P_k$ of the set M such that: 1) $\sum_{i \in P_j} s_i + H \leq W$, $\forall j = 1 \cdots k$ and 2) each message $m_i \in M$ meets its deadline with $\max_{i \in P_j}\ a_i \leq \min_{i \in P_j}\ d_i$. A solution is optimal if it has minimal k.

The SMMC problem can also be stated as a $0-1$ integer linear program (ILP) as follows:

$$\text{minimize}\ \ k = \sum_{i=1}^{n} y_i \tag{1}$$

subject to constraints

$$\begin{aligned}
\sum_{j=1}^{n} s_j x_{ij} + H \leq W y_i, &\quad \forall i \in \{1 \cdots n\} \\
\max\ a_j x_{ij} \leq \min\ d_j x_{ij}, &\quad \forall i \in \{1 \cdots n\}, j \in \{1 \cdots n\} \\
\sum_{i=1}^{n} x_{ij} = 1, &\quad \forall j \in \{1 \cdots n\} \\
y_i \in \{0, 1\}, &\quad \forall i \in \{1 \cdots n\} \\
x_{ij} \in \{0, 1\}, &\quad \forall i \in \{1 \cdots n\}, \forall j \in \{1 \cdots n\}
\end{aligned}$$

where $y_i = 1$ if packet i is used and $x_{ij} = 1$ if message j is put into packet i.

In the formulations above, the term deadline refers to the local deadline for a message at the DCU by which a particular message must be picked up for the packet creation and transmission over the network. This local deadline can be set by subtracting away an estimate of processing delay at the DCU and the network delay over the backhaul from the end-to-end deadline specification of an application for messages. We will discuss and incorporate the impact of processing and network delays later in Section V. In the problem definition above, for any set of messages assigned to a packet, none of the messages in the packet will miss their local deadlines at the DCU if the arrival times of all messages are at least some value ϵ before the first expiring deadline value among all messages of that set. This value ϵ could be set to the maximum processing delay to be encountered at the DCU in forming a packet and could be an input to the problem; more discussion about estimation of processing delays will be presented in Section V.

III. Algorithms for the SMMC Problem

A. SMMC Hardness Result

To prove that the SMMC problem is NP-complete we first show that SMMC is in NP, or in other words, has a polynomial time verifier. An instance of a solution to the SMMC problem is an integer number of packets k and a feasible k-partition $P_1 \cup P_2 \cup \cdots \cup P_k$ of the set of messages M. Such an instance can be verified in polynomial time in terms of the input consisting of the following fields `<message identifier, arrival time, deadline, message size, header size,` W`>` for n messages. Further, in polynomial time (in terms of input length) we can check that each message falls in exactly one of the k partitions/packets, and that each packet meets the condition of having its total size less than or equal to W. We can further check in polynomial time if any message in the packet will miss its local deadline. Thus, we can verify whether a given instance is a solution to SMMC in polynomial time, and hence, SMMC $\in$ NP.

To prove that the SMMC problem is NP-hard we reduce the known NP-complete bin packing problem [16] to the SMMC problem. These problems have many similarities but differ in terms of the notion of arrival times and deadlines for the SMMC problem. The bin packing problem takes as input a set of n' items $I = \{\mathrm{it}_1, \ldots, \mathrm{it}_{n'}\}$ of sizes $S' = \{s'_1, s'_2, \ldots, s'_{n'}\}$ and a set of bins $B = \{b_1, \ldots, b_{k'}\}$ each of size W'. An assignment of items to bins is sought that minimizes the number of bins k' into which all items are packed. That is we seek a k'-partition $B_1 \cup B_2 \cup \cdots \cup B_{k'}$ of the set of items I.

We will transform an instance of the bin packing problem to that of the SMMC problem as follows. For each item i in I, add dummy variables $A' : a'_i = 0$, and $D' : d'_i = \infty$. This transformation can be trivially done in polynomial time (in terms of input length) and the modified instance used as an input to the SMMC problem with $M = I$, $S = S'$, $D = D'$, $A = A'$, $W = W'$, and $P = B$.

Any resulting solution from the SMMC problem can be transformed back to a solution for the bin packing problem as follows. A solution to the SMMC problem gives an integer k and a k-partition of M that maps individual messages to specific concatenated packets. We can take this solution and apply the following transformation: $k' = k$ and $B_i = P_i$, $i = 1 \cdots k$. This transformation gives the required solution assignment for the bin packing problem and can be easily done in polynomial time again.

Theorem 1: SMMC is NP-complete.

Proof: By transforming (in polynomial time) any input instance of the bin packing problem to that of an SMMC problem, and the resulting solution of the SMMC problem back to bin packing problem, we have thus reduced bin packing to SMMC. Thus, SMMC is an NP-hard problem. And since we had proved SMMC $\in$ NP earlier, we can conclude that SMMC is NP-complete. ■

The problem as stated so far is an offline version where all packet arrival times and deadlines are known beforehand and the DCU needs to solve the problem looking forward at the entire window of messages that could arrive over duration T.

TABLE II
Proposed Concatenation Heuristics

Algorithm	Description
EDF-DKB	Inserts deadline messages as much as possible inside the packet and the remaining space will be filled through knapsack selection over best-effort messages that have been queued.
EDF-SDKB	Only a single deadline message sits inside the packet with any available space filled with non-deadline messages in the non-deadline queue through knapsack selection.
EDF-FCFS	Messages will be placed in the packet according to their arrival sequence from a common queue of deadline and non-deadline messages on a first-come first-served basis.
EDF-KN	Messages are chosen from a common pool of deadline and best-effort messages selected through the knapsack algorithm.
EDF-KDKB	A sequence of knapsack selections first on all queued deadline messages and then over the queued best-effort messages if needed to fill the packet.
EDF-KBKD	Reverse order of knapsack process in EDF-KDKB working first on the queued best-effort messages and then on the deadline messages if needed.

This problem can occur in practice when all message types and their arrival times are known deterministically, for example, when all messages are scheduled deterministically. However, in most cases the problem will be an online one with stochastic message types and arrivals where the DCU will only have access to those messages (with their arrival time and deadlines) that have reached the DCU and are waiting to be concatenated before being sent out over the backhaul. Thus, any proposed heuristics will need to perform in an online fashion.

B. Heuristics

Due to the proven hardness of the SMMC problem, in this paper, we develop online heuristic-based algorithms for solving the SMMC problem. Our heuristic solution approach is to rely on earliest deadline first (EDF) scheduling where a concatenated packet is created at the DCU starting with a message within a specific threshold of its deadline and then filled with other messages so as to maximize the packet size that can be sent out. Proposed heuristics differ in terms of what other messages they decide to fill in the concatenated packet in addition to the message whose deadline is about to expire.

Six different heuristic-based algorithms are proposed for scheduling of messages at a DCU for the SMMC problem as listed in Table II. All six algorithms initiate creating a packet when one of the local message deadlines at the DCU is about to expire; they differ in terms of what other messages (in addition to the message whose deadline is about to expire) are put in the packet being sent out. In all six schemes, a classifier module checks the arrived messages to see whether they are best-effort or have a specific deadline (if the selected heuristic needs to differentiate between them). Two different queues are formed based on the classification done. All deadline messages are kept in a priority queue sorted by earliest deadline. It is assumed there are two queues in the system, one for the messages with specific delay objective and another for those without a delay objective (the best effort messages). If no classification is required then all arrived messages will be sorted and placed in a single buffer. All of the proposed heuristics (except EDF-FCFS) employ the 0−1 knapsack algorithm [16] to decide which messages to fit into the packet among the various options available. More details of the implementation of our proposed heuristics and associated pseudocode can be found in [17].

C. Reference Algorithms

1) EDF-Based Integer Linear Programming Formulation: To get a solution for the SMMC problem one can use mathematical optimization algorithms. We have formulated the SMMC problem as a mixed-ILP which optimally schedules the remaining messages in addition to the EDF message to begin a packet with index. The problem is formulated as follows for a packet with index i:

$$\text{maximize} \;\; P_i = \sum_{j=1}^{n_t} s_j x_{ij} \tag{2}$$

subject to constraints

$$\sum_{j=1}^{n_t} s_j x_{ij} + H \leq W$$
$$x_{ij} \in \{0, 1\}$$

where $x_{ij} = 1$ if message j is put into packet i. In the formulation above, n_t ($n_t \leq n$) is the set of messages queued at the DCU and available for concatenation at time t ($t \leq T$). Any messages that are found to not meet deadline constraints are forwarded immediately with no concatenation process applied. This formulation is different from 1 in that it is EDF-based and message deadlines are not a constraint as messages closest to their deadlines are selected and sent out before their deadlines occur. This formulation tries to fit in as many messages as possible (among those available) in a packet to be sent out. The given constraint specifies the maximum packet size that can be sent over the backhaul technology with a specific MTU size. The drawback of this approach in practice (as opposed to our heuristics) is the brute force nature of this ILP solution procedure which makes it practically infeasible for real-time applications and those that involve large-scale data.

2) Theoretical Optimum: This method is theoretically the minimal number of packets that needs to go out of a DCU for a given number of messages generated from the smart meters over a period of time. This value is not constrained by arrival times or deadlines of messages; it is computed through the equation $\lceil (\sum_{i=1}^{n} s_i/\text{MTU} - H) \rceil$ where n is the total number of arrived messages during a time interval, and s_i is the size of a message i. MTU and header size H are the parameters defined according to the backhaul technology. Although this solution is not feasible in practice, it gives a theoretical reference for the performance evaluation of any SMMC algorithms, not limited to EDF-based heuristics.

IV. Evaluation

A. Methodology

We outline below more details about the simulation environment, message arrival process, and distribution of various message types.

1) Simulation Environment: A discrete-event simulator was developed using MATLAB to evaluate the proposed heuristic-based algorithms and compare to the reference algorithms. The network topology consisted of a group of smart meters generating messages as a poisson process and sending messages to the DCU to be routed to the control center.[1] Due to the assumption of individual meter message generation as a poisson process, we can sum the individual average message generation rates to get an cumulative average arrival rate at the DCU of λ which is used as a parameter in our simulations. We have considered three different λ values of 0.1, 0.5, and 1 at the DCU which would correspond to 90, 450, and 900 smart meters sending one message on average every 15 min. The service capacity of the DCU is considered to be infinite; however, we do study the impact of processing delays in the following section.

TABLE III
PREDEFINED MESSAGE ARRIVAL DISTRIBUTIONS

Distribution	Description
Uniform ($\alpha = 1$, $\beta = 1$)	The traffic would have almost equal percentage of all message types.
More smaller ($\alpha = 2.8$, $\beta = 1.9$)	Most of the arrived messages are of the smaller size of message types.
More larger ($\alpha = 0.18$, $\beta = 0.25$)	There is higher percentage of large message size and very few numbers of small size messages.
More deadline ($\alpha = 1$, $\beta = 1.8$)	Most of the times there are incoming messages with deadline restriction.
More best-effort ($\alpha = 2.5$, $\beta = 0.5$)	There are very few numbers of messages with a deadline and so many best-effort messages.

2) Message Types Distribution: During a day, different types of the messages may be exchanged between smart meters and the utility control center through the AMI. In our evaluations we have considered all seven basic types of messages first reported in [5]. Based on geographic location, power distribution infrastructure, and utility preferences, the transmission of messages could come from different distributions of these basic message types which will have an impact on the performance of our proposed heuristics. In our evaluations we used different Beta distributions across these message types by varying shape parameters $\alpha > 0$ and $\beta > 0$.

For our experiments, we generated five different message type distribution using the shape parameters mentioned in Table III to test the performance of our proposed algorithms.

B. Simulation Results

Simulations were conducted with 100 runs and the mean value plotted in results shown along with 95% confidence intervals. Each scheme was evaluated in terms of the overall reduction in bytes of data transmitted out into the backhaul network by the DCU as compared to the overall incoming data in bytes from smart meters, including all headers. Each packet header was assumed to be of a fixed size of 50 bytes corresponding to the 40–60 bytes range for TCP and IP headers. Fig. 3 displays the output of our proposed algorithms and reference algorithms over five message types distributions with 95% confidence intervals. Results are shown for packet arrival rates at the DCU of $\lambda = 0.1, 0.5$, and 1. It can be seen that

[1]Reference [18] supports this assumption that smart meters message generation can be modeled as a poisson process.

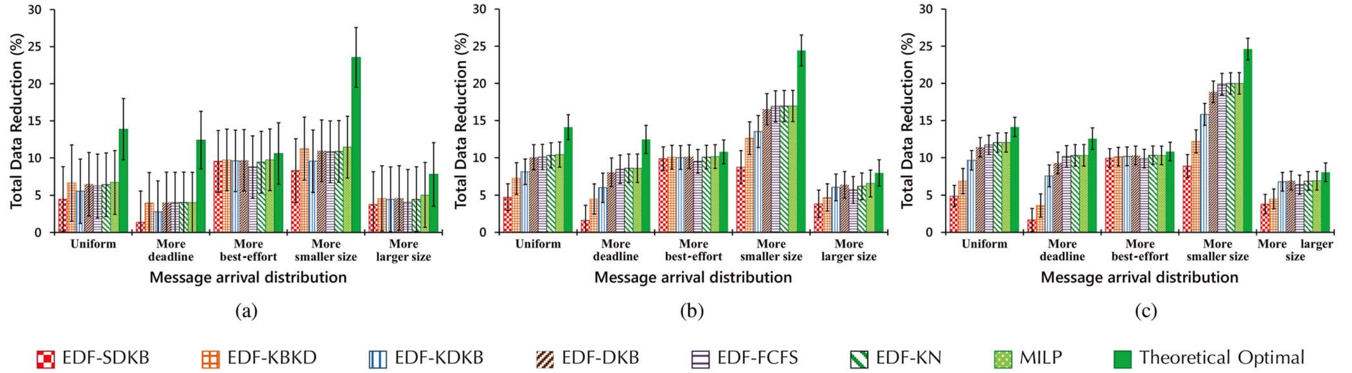

Fig. 3. Overall data reduction percentage using proposed heuristics over different message arrival rate and message type distributions. (a) $\lambda = 0.1$. (b) $\lambda = 0.5$. (c) $\lambda = 1$.

overall data volume reduction varies from 5%–25% depending on message type distribution, message arrival rate at DCU, and specific algorithm used. Three questions answered are as follows.

1) How do the Proposed Heuristics Stack Up Against Each Other and Reference Algorithms?: Taking a look at the bar charts in Fig. 3 one can observe that the algorithm EDF-KN has the best performance among all other heuristics and comes very close to the performance of the EDF-based ILP across all λ and message type distributions. This is due to the fact that EDF-KN is using a common pool of messages whether they be deadline or best effort, giving more options to maximize packet size before it is sent out. Since typically there are enough queued messages before a deadline reaches, the algorithm has a good collection of options to maximize the packet before sending it out. It can be noticed that in general EDF-based approaches do well compared to theoretical volume reduction, where the latter increases with MTU size and decreased with the size of H.

2) What is the Impact of Message Type Distribution?: The uniform distribution of all message types serves as the reference case to compare other distributions. For the more deadline case with a majority of all messages having deadlines, overall data volume reduction is smaller for all algorithms. Presence of more messages with deadlines than best-effort necessitates packets to be sent out of the DCU without having the luxury of waiting for the right combination to maximize packet size. However, when there are more best-effort messages present, algorithms can wait longer before being forced to send out packets; this allows each packet to be larger, and hence reduces packet overheads. The case for more smaller size messages is similar to the more deadline message case in that it helps reduce packet overheads significantly through concatenation as header sizes are comparable to data sizes. Smaller messages are also easier to pack into a packet. Conversely, the more larger messages case results in greater difficulty to fill messages into a packet; also larger underlying message sizes already have a reduced overhead making much improvements through concatenation difficult.

3) What is the Impact of λ?: The value of λ signifies the packet arrival rate at the DCU; hence, larger values indicate that more messages are arriving at the DCU increasing opportunities for a concatenation algorithm to find a best fit of messages in an outgoing packet from the DCU to reduce overall protocol overhead. The EDF-KN data volume reduction approaches very close to that of even the theoretically optimal solution with increasing λ. Thus, greater the rate of packet arrivals, the proposed EDF-based concatenation algorithm over a common queue of messages maximizes the reduction in data volume.

V. Impact of Network and Processing Delays

Network delays between the DCU and the utility control center, and processing delay at the DCU itself are two factors we had assumed to be negligible in the results presented so far. The magnitude of these delays may not be negligible in all practical cases, and can cut down the amount of time a DCU can wait to maximize the size of outgoing packets sent out. Thus, there will be a direct correlation between network and processing delays on the ability of a DCU to reduce protocol overhead. An interesting challenge here is that the DCU cannot accurately predict these delays beforehand; each concatenated packet will suffer variable network and processing delays due to many factors related to number of messages processed and characteristics of the communication backhaul. Thus, the DCU needs to rely on an estimate of network and processing delays it needs to budget into computing the local deadline of each message. An overestimate will reduce the amount of time a DCU will have to wait and concatenate a large packet; an underestimate on the other hand can mean some messages will miss their deadlines. This section describes how such delays can be estimated and what impact it will have on data volume reduction through message concatenation.

A. Estimation of Network and Processing Delays

To estimate the processing delay, we need to break it into the major individual components that cause delay. These components are a follows.

1) *Concatenation Delay:* The time required to put all selected messages into a packet and add a common header.

TABLE IV
HEURISTICS PROCESSING TIME CALCULATIONS

Heuristic	Processing Delay
EDF-FCFS	$C_C + C_S$
EDF-KN	$C_C + C_S + C_K(n)$
EDF-DKB	$C_C + C_S + C_K(n_1)$
EDF-SKB	$C_C + C_S + C_K(n_2)$
EDF-KDKB	$C_C + C_S + C_K(n_1) + C_K(n_2)$
EDF-KBKD	$C_C + C_S + C_K(n_2) + C_K(n_1)$

2) *Knapsack Delay:* The time required by some of the schemes that use a knapsack operation to select messages from a queue of messages.
3) *Sorting Delay:* The time required to maintain the queue sorted in terms of earlier deadlines.

These components are present in each heuristic in possibly different ways based on the nature of the algorithm. Table IV summarizes how each of these components (C_C, C_S, and C_K time costs for concatenation, sorting and selection through knapsack, respectively) sum up to the total processing delay for each heuristic scheme. These schemes operate on either a single common queue of n items, or one of two queues (with sizes n_1 and n_2) having deadline and nondeadline messages, or both queues one after the other. The next step was to populate realistic values into the processing delay estimation model. For this, we measured actual processing delays when executing each of the three operations: 1) concatenation; 2) knapsack selection; and 3) keeping a sorted queue. These values were computed on a Dell Optiplex 64-bit PC with a 2-core 2.8 GHz CPU and 5 GB RAM for a full range of values of n from 1 to 1000 to study all possible queue sizes we are likely to encounter for message arrival rates used in the evaluations in Section IV.[2] By populating these values for a given n in the processing delay model presented in Table IV, the DCU could easily construct an estimate.[3]

B. Evaluation Results

Here we re-evaluate our proposed heuristic-based algorithms with varying values of network and processing delays, and study the impact on achievable reductions in protocol overhead. For these evaluations we have chosen the EDF-KN heuristic, one of the better performing heuristics among those evaluated in Section IV-B. Fig. 4 presents the results for $\lambda = 1$ and shows the protocol overhead reduction achieved with varying values of network and processing delays, including the case where such delays are set to nil. In addition, to further explore the lower limits of possible benefits of message concatenation, we experiment with deadline values half and quarter the amount of that used in our evaluations in Section IV. The impact of tighter deadlines will be similar to that of additional network and processing delays, with both factors essentially reducing the time the DCU has to concatenate messages into larger packets.

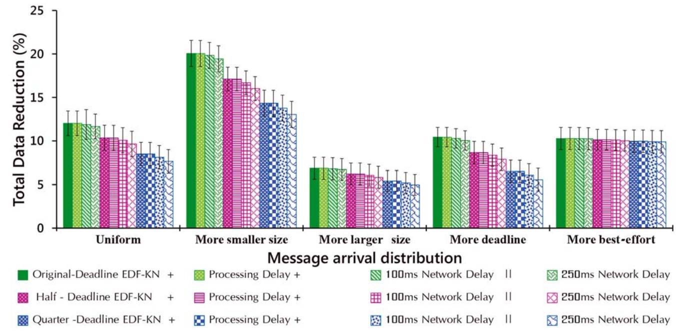

Fig. 4. Data reduction trend versus delay addition.

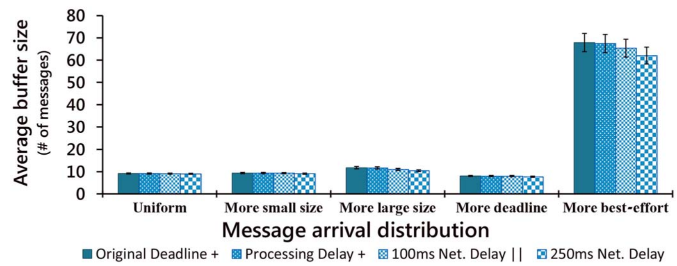

Fig. 5. Average buffer size versus delay addition.

The results in Fig. 4 show that as processing and network delays increase, the percentage overhead reduction decreases. Similarly, as deadlines get tighter, the data volume reduction achievable reduces. Even for such extreme cases considered, there is at least a 5% reduction in data volume possible. The biggest impact of network and processing delays, or tighter deadlines is with the "more deadline" message distribution with a greater fraction of messages needing to be concatenated and sent out quickly. The smallest impact of delays or tighter deadlines is seen for the "more best-effort" case where most messages are not hard-pressed to meet deadlines.

A more accurate depiction of what happens inside the DCU can be seen by studying the average queue or buffer size for various message type distributions for estimated processing delays and varying network delay values of 100 and 250 ms. A similar trend can be expected for tighter deadline values. As Fig. 5 confirms, the more deadline message distribution has the smallest average queue size, implying that messages do not stay in the buffer for long periods. The more best-effort message distribution at the other extreme results in the largest average queue size implying messages stay in the buffer for a much longer duration. A large average queue size does add additional processing delay; however, for the more best-effort case, there are few messages with deadlines that are impacted by the larger processing delays. For all the other schemes, evident from the results, the average queue size stays small enough to not adversely impact data volume reduction.

[2]We assume that when our algorithms are deployed, an estimate can be recalculated for the specific system employed in the DCU as opposed to using the estimates discussed here. DCUs on the market can have high-processing capabilities as described in [19] and we expect the values used in this paper to over-estimate actual processing delays.

[3]Due to space restrictions in this paper, we refer the reader to [20] for a description of how network delays can be predicted with an exponentially weighted moving average over a sliding window of previously seen delays. We will study the impact of various possible network delays to assess the impact on benefits of message concatenation in evaluations that follow.

VI. Data Volume With Lossy Links

Another practical aspect that needs to be considered is the impact of lossy backhaul links on the large concatenated packets expected to be sent out from the DCU by proposed heuristic-based algorithms. Larger packets will typically suffer more retransmissions (and thus adding to data volume transported) when sent through networks with a fixed bit-error rate (BER) due to their larger size. Thus, it is imperative to explore the impact of various backhaul technologies, each with different BER characteristics, on benefits of message concatenation.[4]

A. Theory

The most important factor in analyzing the impact of lossy networks is considering the BER of the technology being used. The transmission BER is the number of detected bits that are incorrect before error correction, divided by the total number of transferred bits (including redundant error codes). Different communication technologies have different BER. The goal here is to translate a given BER for a technology and estimate the corresponding data volume reduction ratio. Let e_b be the BER of a given technology. A packet is declared incorrect if at least one bit is erroneous. Thus, for a packet of size L bits, the resulting packet error rate (PER) of the technology, e_p, then is $e_p = 1 - (1 - e_b)^L$. Let D be the volume of data in bytes (including payload and control overhead) that would have been sent over the backhaul in a time period T when message concatenation is not employed. Let D' be the volume of data sent (again including payload and control overhead) over the backhaul after message concatenation. With a PER of e_p and e'_p, respectively, the corresponding data volume sent through the backhaul will be $(1 + e_p)D$ and $(1 + e'_p)D'$, respectively. Thus, the data volume reduction ratio ρ with a lossy backhaul can be computed as

$$\rho = \frac{(1+e_p)D - \left(1+e'_p\right)D'}{(1+e_p)D}. \tag{3}$$

With larger packet sizes $e'_p > e_p$, thus reducing the data volume reduction ratio as compared to the case when lossiness of the backhaul network is ignored.

B. Numerical Evaluation

The technologies for the backhaul considered are fiber optic, WiMAX, and 3G cellular; these three technologies are currently commonly used to connect the AMI at the customer to the backbone network and tend to be lossier than the core network. We picked BER values for these technologies based on known ranges in [21]–[23] to study the impact of message concatenation algorithms. The BER values e_b used in the following evaluation were 5E-07, 3.16E-06, and 7.5E-06 for fiber optic, WiMAX, and 3G technologies, respectively. For each technology, we computed PERs e_p using the equation above.

[4]Due to space restrictions, we do not explore the analogous issue of packet loss due to network congestion; the eventual impact on the benefits of message concatenation is expected to be similar regardless of the underlying reason for packet loss

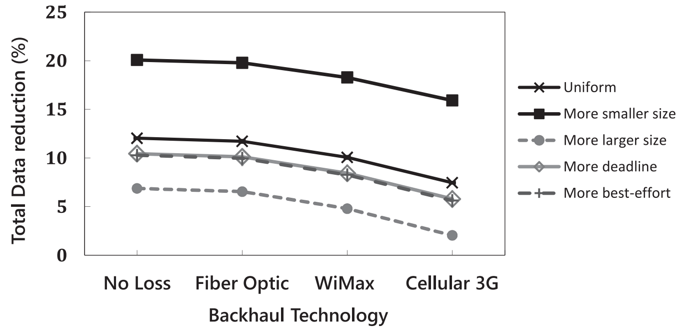

Fig. 6. Data reduction savings versus different backhaul technologies.

For the case with no message concatenation, we considered an average packet size of 100 bytes ($L = 800$ bits) in computing a PER of e_p; for the case with concatenation, we used a packet size of 1000 bytes ($L = 8000$ bits) to compute e'_p which is roughly the average size of concatenated packet seen in our simulations from the earlier sections. Finally, using 3, we computed ρ for each of the three technologies with D and D' computed based on our simulations earlier in Section IV for the EDF-KN scheme with a message arrival rate of $\lambda = 1$.

It can be seen from Fig. 6 that for even the most lossy technology considered (3G) with worst-case BER characteristics chosen, data volume reduction with message concatenation only falls by 3%–4% compared to the reference ideal BER case. Thus, the benefits of message concatenation seems to hold up for the most commonly used technologies. These results are likely to be better with the use of forward error correction techniques employed to minimize packet loss.

VII. Case Study of Practical Benefits of Proposed Algorithms

With many hundreds of thousands of customers in a geographic location, utilities will be thus sending data in the order of Mb/s to Gb/s through their backhaul networks connecting to control centers. This scale of data flow through AMI networks is also supported by [24]–[26]. This section presents a case study of actual data rates flowing through neighborhood networks of different sizes and how it may impact a given backhaul communication technology and the applicability of proposed data concentration algorithms.

Assume a power system topology with a feeder connecting to 1350 customers in an area with 450 distribution transformers, with one transformer connecting to three customer smart meters. This chosen topology is typical of for a suburban area in the U.S. (see [27]). A logical communications network overlaid on the physical topology of this distribution system topology could be as follows. Based on the manner in which the communications network is organized, its communication range, and the customer meter density, x smart meters could be connected to a DCU. For the topology assumed, x could take on any values from 1 to 1350. The total number of DCUs required would depend on the value of x. The DCUs are then further connected through a backhaul to the communications network. With x meters each sending a message every y seconds, the average data arrival rate at each DCU will be x/y messages per second. For message sizes averaging 350 bytes and a 50-byte header (typical sizes from [5]), this amounts to a data rate of $3.2x/y$ Kb/s at each DCU employed. For $x = 450, 900, 1500$ and for $y = 900$ s (15 min intervals), this amounts to data rates per DCU of 1.6, 3.2, and 4.8 Kb/s. For more fine-grained data collections in the future for analysis (e.g., as motivated in [28]) or just applications such as EV load control and appliance-level load monitoring, y could be of the order of few seconds. For 10 s intervals, this results in data rates of 144, 288, and 432 Kb/s for $x = 450, 900$, and 1350, respectively.

A technology like power line communications (PLC) can only support data rates in the order of Kb/s [29]. Thus for neighborhood deployments of the order of 500–1500 smart meters, with a low-bandwidth technology like PLC, it is imperative that data volume through such backhaul links be managed carefully. Other higher bandwidth backhaul links such as cellular, WiFi, and WiMAX can support higher data rates (at higher costs) and will be less stressed by smart meter deployments. With electric utilities either leasing communications capacity from telecom companies, or building their own telecommunications networks and then leasing capacity to recuperate costs, they will benefit from reducing the amount of data sent through their networks regardless of the scale of a smart meter deployment and bandwidth of communication links. A 20% reduction in data volume (as can be achieved by the proposed heuristics in this paper) should translate to a similar reduction in network infrastructure costs under a scenario of per byte capacity costs. Such reduction in costs is expected to also benefit all customers, whether they are equipped with smart meters or not. As the penetration of smart meters increases, the applicability of this paper will keep increasing with more benefits for greater traffic volumes as found in our results earlier in this paper. The Federal Energy Regulatory Commission survey in 2012 [30] indicated AMI penetration to be about 23% (a 14% increase over 2010 levels) and is expected to have increased at a similar rate since then.

A scenario where the applicability of the proposed data concentration approach would be reduced is if network capacity is not metered per byte of data transported, but instead is a fixed capacity cost, and if the smart meter deployments are small enough to not stress deployed networks. The data flow analysis in the previous paragraph shows that in such a case, for neighborhoods as small as 500–1500 smart meters connected to a single DCU, the proposed data concentration schemes may be useful only if a low-bandwidth technology like PLC is used for the backhaul. However, if multiple such neighborhoods are clustered together behind a single data concentrator (with appropriate network topology configurations), the data concentration schemes would still be useful for even high-bandwidth technologies. For larger number of meters, such as 3000 and above, data generated at (10 s interval collection) would be of the order of Mb/s and can stress higher bandwidth links and be very useful even for those links.

VIII. CONCLUSION

This paper demonstrated that message concatenation algorithms can be an important element of data concentrators deployed in smart grids to solve the looming challenge of transporting massive data volumes through last mile bandwidth-constrained backhaul networks. Effective message concatenation algorithms at DCUs (such as the EDF-KN algorithm proposed in this paper) were shown to be able to reduce overall data volume by 10%–25% for each DCU. This reduction was achieved just by a reduction in protocol overhead with no compression of the original data sent by smart meters; this provides enough motivation to develop additional data concentration mechanisms at DCUs that also act on the payload of messages. Another direction of future work is to look at how concatenation can be done at multiple levels of the communications network, not limited to just the first hop from smart meters.

REFERENCES

[1] National Energy Technology Laboratory. (Feb. 2008). *Advanced Metering Infrastructure*. [Online]. Available: http://www.netl.doe.gov/smartgrid/refshelf.html, accessed Oct. 27, 2011.

[2] U.S. Department of Energy. (2008). *What the Smart Grid Means to Americans*. [Online]. Available: http://www.doe.gov/sites/prod/files/oeprod/DocumentsandMedia/ConsumerAdvocates.pdf

[3] D. Bernaudo *et al.*, "SmartGrid/AEIC AMI interoperability standard guidelines for ANSI C12.19/IEEE 1377/MC12.19 end device communications and supporting enterprise devices, networks and related accessories," Assoc. Edison Illuminating Companies, Birmingham, AL, USA, Meter Service Tech. Commit. Rep. Ver. 2, Nov. 2010.

[4] E. E. Queen, *A Discussion of Smart Meters and RF Exposure Issues*, Edison Elect. Inst., Washington, DC, USA, Joint Proj. EEI AEIC Meter Commit., Mar. 2011.

[5] W. Luan, D. Sharp, and S. Lancashire, "Smart grid communication network capacity planning for power utilities," in *Proc. IEEE PES Transmiss. Distrib. Conf. Expo.*, New Orleans, LA, USA, Apr. 2010, pp. 1–4.

[6] M. Allalouf, G. Gershinsky, L. Lewin-Eytan, and J. Naor, "Data-quality-aware volume reduction in smart grid networks," in *Proc. IEEE Int. Conf. Smart Grid Commun. (SmartGridComm)*, Brussels, Belgium, 2011, pp. 120–125.

[7] *High-Level Smart Meter Data Traffic Analysis*, Document ENA-CR008-001-1.4, Energy Networks Association, London, U.K., May 2010. [Online]. Available: http://www.energynetworks.org/modx/assets/files/electricity/futures/smart_meters/ENA-CR008-001-14_DataTrafficAnalysis_.pdf

[8] Greentechgrid. (Feb. 15, 2015). *Arcadian's Smart Grid: Licensed Spectrum Network to Own or Rent*. [Online]. Available: http://www.greentechmedia.com/articles/read/arcadians-utility-offering-licensed-spectrum-to-own-or-rent

[9] M. Kennedy. (Feb. 15, 2015). *Leveraging Investment in Fiber Optic Communications, IEEE Smart Grid*. [Online]. Available: http://smartgrid.ieee.org/june-2011/105-leveraging-investment-in-fiber-optic-communications

[10] PRIME Alliance. (Aug. 8, 2013). *PoweRline Intelligent Metering Evolution*. [Online]. Available: http://www.prime-alliance.org

[11] A. F. Snyder and M. T. G. Stuber, "The ANSI C12 protocol suite-updated and now with network capabilities," in *Proc. Power Syst. Conf. Adv. Meter. Protect. Control Commun. Distrib. Resour.*, Clemson, SC, USA, Mar. 2007, pp. 117–122.

[12] R. Rajagopalan and P. Varshney, "Data-aggregation techniques in sensor networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 8, no. 4, pp. 48–63, Mar. 2006.

[13] S. Zhu, W. Wang, and C. V. Ravishankar, "PERT: A new power-efficient real-time packet delivery scheme for sensor networks," *Int. J. Sens. Netw.*, vol. 3, no. 4, pp. 237–251, Jun. 2008.

[14] S. Hariharan and N. Shroff, "Maximizing aggregated revenue in sensor networks under deadline constraints," in *Proc. IEEE Decis. Control (CDC)*, Shanghai, China, 2009, pp. 4846–4851.

[15] S. Hariharan and N. B. Shroff, "Deadline constrained scheduling for data aggregation in unreliable sensor networks," in *Proc. WiOpt*, Princeton, NJ, USA, 2011, pp. 140–147.

[16] T. H. Cormen, C. Stein, R. L. Rivest, and C. E. Leiserson, *Introduction to Algorithms*, 2nd ed. New York, NY, USA: McGraw-Hill, 2001.

[17] B. Karimi, V. Namboodiri, and M. Jadliwala, "On the scalable collection of metering data in smart grids through message concatenation," in *Proc. 4th IEEE Int. Conf. Smart Grid Commun. (SmartGridComm)*, Vancouver, BC, Canada, Oct. 2013, pp. 318–323.

[18] H. Li, Z. Han, L. Lai, R. Qiu, and D. Yang, "Efficient and reliable multiple access for advanced metering in future smart grid," in *Proc. IEEE Smart Grid Commun. (SmartGridComm)*, Brussels, Belgium, 2011, pp. 440–444.

[19] S. Evanczuk. (Mar. 2013). *Data Concentrators Combine AFEs, MCUs, and Radios to Key Smart Grid Efficiency*. [Online]. Available: http://www.digikey.com/us/en/techzone/energy-harvesting/resources/articles/data-concentrators-combine-afes-mcus-radios.html

[20] V. Namboodiri and L. Gao, "Energy-efficient VoIP over Wireless LANs," *IEEE Trans. Mobile Comput.*, vol. 9, no. 4, pp. 566–581, Apr. 2010.

[21] O. V. Sinkin, "Calculation of bit error rates in optical fiber communications systems in the presence of nonlinear distortion and noise," Ph.D. dissertation, Faculty Grad. School, Univ. Maryland, College Park, MD, USA, 2006.

[22] H. Kaur and M. L. Singh, "Bit error rate evaluation of IEEE 802.16 (WiMAX) in OFDM system," *Int. J. Comput. Appl.*, vol. 40, no. 12, pp. 10–13, Feb. 2012.

[23] P. Geremia, P. Bertrand, S. Tomas, and M. Ahnoff, "TMS320c6416 coprocessors and bit error rates," Texas Instruments, Dallas, TX, USA, Tech. Rep. SPRA974, Nov. 2003.

[24] D. Alahakoon and X. Yu, "Advanced analytics for harnessing the power of smart meter big data," in *Proc. IEEE Int. Workshop Intell. Energy Syst. (IWIES)*, Vienna, Austria, Nov. 2013, pp. 40–45.

[25] M. Ringwelski, C. Renner, A. Reinhardt, A. Weigel, and V. Turau, "The Hitchhiker's guide to choosing the compression algorithm for your smart meter data," in *Proc. IEEE Int. Energy Conf. Exhibit. (ENERGYCON)*, Florence, Italy, Sep. 2012, pp. 935–940.

[26] W. Luan, D. Sharp, and S. LaRoy, "Data traffic analysis of utility smart metering network," in *Proc. IEEE Power Energy Soc. Gen. Meeting (PES)*, Vancouver, BC, Canada, Jul. 2013, pp. 1–4.

[27] M. Oens and C. Lange, *Improvements in Feeder Protectionproviding a Primary and Backup Relay System Utilizing One Relay Per Feeder*, Schweitzer Eng. Lab. Inc., Pullman, WA, USA, 2013. [Online]. Available: https://www.selinc.com/WorkArea/DownloadAsset.aspx?id=3417

[28] B. McCracken, M. Crosby, C. Holcomb, S. Russo, and C. Smithson, "Data-driven insights from the nations deepest ever research on customer energy use," Pecan Res. Inst., Austin, TX, USA, 2013. [Online]. Available: http://www.nhpci.org/publications/NHPCW-hitepaper-Making-Sense-of-Smart-Home-final-20140425.pdf

[29] M. Hoch, "Comparison of PLC G3 and PRIME," in *Proc. IEEE Int. Symp. Power Line Commun. Appl. (ISPLC)*, Udine, Italy, Apr. 2011, pp. 165–169.

[30] "Assessment of demand response and advanced metering," Fed. Energy Regul. Comm., Washington, DC, USA, Tech. Rep., Dec. 2012. [Online]. Available: http://www.ferc.gov/legal/staff-reports/12-20-12-demand-response.pdf, accessed Sep. 1, 2014.

Babak Karimi (S'10) received the M.S. degree in information technology from the Amirkabir University of Technology, Tehran, Iran, in 2008, and the M.S. degree in computer networking from Wichita State University, Wichita, KS, USA, in 2012, where he is currently pursuing the Ph.D. degree focusing on the application of wireless communications in the smart grid.

He is currently a Senior Network Engineer with Autodesk, Inc., San Francisco, CA, USA. His current research interests include designing architecture for smart grid communications, solving problems related to advanced metering infrastructure, and data concentration along with its security and privacy issues.

Vinod Namboodiri (M'04) received the Ph.D. degree in electrical and computer engineering from the University of Massachusetts Amherst, Amherst, MA, USA.

He is currently an Associate Professor with the Department of Electrical Engineering and Computer Science, Wichita State University, Wichita, KS, USA. His current research interests include designing algorithms and protocols for energy-intelligent and sustainable computing, and designing an effective and scalable communications architecture for smart electric grids.

Prof. Namboodiri has served on the Technical Program Committee of IEEE INFOCOM, IEEE SmartGridComm, IEEE GLOBECOM, IEEE ICC, IEEE IPCCC, and IEEE GREENCOM. He is an Active Reviewer for numerous journals and conferences in the mobile computing, green computing, and smart grid areas.

Murtuza Jadliwala (M'04) received the Ph.D. degree from the State University of New York at Buffalo, Buffalo, NY, USA, in 2008.

He is currently an Assistant Professor with the Department of Electrical Engineering and Computer Science, Wichita State University, Wichita, KS, USA, where he directs the Security, Privacy, Trust, and Ethics in Computing Research Labratory. His current research interests include application, network, and cyber-physical system security and privacy enhancing technologies.