Automated Configuration Synthesis for Resilient Smart Metering Infrastructure

Mohammad Ashiqur Rahman[1,*], Amarjit Datta[2], and Ehab Al-Shaer[3]

[1]Department of Electrical and Computer Engineering, Florida International University, Miami, USA
[2]Department of Computer Science, Tennessee Technological University, Cookeville, USA
[3]CyLab Security and Privacy Institute, Carnegie Mellon University, Pittsburgh, USA

Abstract

An Advanced Metering Infrastructure (AMI) comprises a large number of smart meters along with heterogeneous cyber-physical components that are interconnected through different communication media, protocols, and delivery modes for transmitting usage reports or control commands between meters and the utility. Due to misconfigurations or lack of security controls, there can be operational disruptions leading to economic damage in an AMI. Therefore, the resiliency of an AMI is crucial. In this paper, we present an automated configuration synthesis framework that mitigates potential threats by eliminating misconfigurations and keeps the damage limited under contingencies by introducing robustness. We formally model AMI configurations, including operational integrity and robustness properties considering the interdependencies among AMI devices' configurations, attacks or failures, and resiliency guidelines. We implement the model using Satisfiability Modulo Theories (SMT) and demonstrate its execution on an example case study that illustrates the synthesis of AMI configurations satisfying resiliency requirements. We also evaluate the framework on synthetic AMI networks.

1. Introduction

AMI is a core component of a smart grid and it provides two-way communication between smart meters and the energy utility's network (particularly, the headend servers within the utility) through intelligent data concentrators or collectors, which allows energy service providers to monitor and control power consumption remotely [1].These devices communicate with one another using different communication media, protocols, security policies, and data delivery modes. Dependability threats to AMI, especially due to inappropriate configurations and weak resiliency measures, can cause critical operational disruption leading to economic damages [2].

Misconfigurations can cause nontrivial threats to the secure and reliable operation of an AMI system as there are interdependencies among device configurations,

communication and data delivery properties, and mission requirements. Moreover, when an intermediate device (*e.g.*, a collector) or a communication link breaks down, then the data delivery from associated meters to the utility fails if there is no alternative path. On the other hand, due to the limited budget and benefit, it is not possible to deploy so many redundant devices to create alternative communication paths for having resiliency, while it is crucial to achieve a level of robustness with the limited resources. Manual enforcement of the appropriate AMI configurations can be overwhelming and often inaccurate and non-optimal due to high potentiality of human errors. Therefore, there is a pressing need for the automatic synthesis of the AMI architecture, *i.e.*, the topology and devices' configurations, ensuring resiliency (*i.e.*, operational integrity and robustness) of the system.

---

*Corresponding author. Email: marahman@fiu.edu

In this research, we address this need by presenting an automated resiliency configuration synthesis framework for an AMI system. In this framework, we create a logic-based formal model of the AMI topology and devices' configurations, data delivery among the devices, operational integrity invariants, and robustness requirements, according to a set of inputs. The solution to this formal model synthesizes resiliency configurations, including a deployment or placement design for collectors and report (or report request) schedules for meters, collectors, and the headend system. We apply abstraction in modeling smart meters and their association with collectors, which allows the proposed synthesis mechanism to scale with large numbers of smart meters. We implement the framework and illustrate its execution using an example case study. We use SMT to formalize the framework. SMT consists of powerful logic theories that can solve hard constraint satisfaction problems which arise in many diverse areas, including software and hardware verification, test-case generation, scheduling, and planning [3]. We also evaluate the accuracy and scalability of our framework by running it on various synthetic test networks. Partial results of this research has been published in [4], where the operational constraints were considered without the resiliency properties. In a nutshell, this paper presents the following novel contributions:

- The synthesis of AMI configurations is modeled with respect to resiliency constraints that consider operational integrity and robustness in contingencies.
- The resiliency synthesis framework is demonstrated using a case study that illustrates the synthesis of AMI configurations satisfying the resilient operational constraints.
- A detail scalability evaluation of the proposed synthesis framework is performed on various synthetic problems.

The rest of the paper is organized as follows. We discuss the motivation of our research in Section 2. In the next section, we present the synthesis framework. In Section 4, we describe the framework. The evaluation results are presented in Section 5. We briefly discuss about the limitations, extensibility, and deployment of the proposed solution in Section 6. The paper is concluded in Section 7.

2. Background and Challenges

This section discusses necessary backgrounds and research challenges with regards to this work.

### 2.1. Advanced Metering Infrastructure

The typical network structure of an AMI system is shown in Fig. 1 [5, 6]. An AMI system often consists of thousands of smart meters and hundreds of intelligent data concentrators or collectors. A meter reports energy usage data to a specific collector periodically. A collector stores the data received from a group of meters in its buffer and forwards the stored data to a server located at the utility's network. This server is often known as the headend system. Although in some AMI architectures, a meter directly reports energy usage data to the headend system, often data collectors are used to collect and store meter data and later to send the stored data to the headend system when it is required [1, 5–8]. This collector based AMI design gives better manageability by allowing scalable infrastructure design, flexible protocol use, and efficient networking. The collector also forwards control commands and patches from the headend to the meters. A meter is connected to a collector either directly or through another meter. The latter case occurs in a mesh network of meters, where intermediate meters relay the data to the collector. Collectors are connected to a headend usually through a proprietary but often a third party network. The communication mediums among meters, collectors, and the headend system can be power lines, wired, wireless, or cellular, while the communications are performed using TCP/IP or proprietary (e.g., LonTalk [9]) protocols.

Unlike the policy-based Internet forwarding, data deliveries in an AMI network are either time-driven or request-driven and they follow specific schedules. In the time-driven (push) mode a meter or a collector reports data periodically based on a pre-configured delivery schedule, while in the request-driven (pull) mode a meter or a collector reports data only upon receiving a request. In the pull mode, requests are often sent periodically following a schedule [5, 10]. In practice, the push mode is used between meter and collector, while the pull mode is used between collector and headend. For the purpose of successful delivery of data, an AMI system must be configured carefully to synchronize the data delivery without overflowing the network or its devices.

2.2. Causes of Threats on AMI

There are two major causes of threats on AMI: (i) vulnerability due to misconfigurations or weak security controls and (ii) lack of preparedness against attacks [11, 12]. It is well documented that configuration errors cause 50%-80% of vulnerabilities in cyber infrastructure [13]. Improper data scheduling due to misconfiguration can cause data loss by overflowing the communication bandwidth or the collector's storage capacity. Let us provide an example of such a misconfiguration scenario. Although the values used in this example are synthetic, they are motivated from [5, 7, 14, 15]. A collector receives reports from 100 meters of two types. Each meter of one kind has a sampling rate of 18 KB
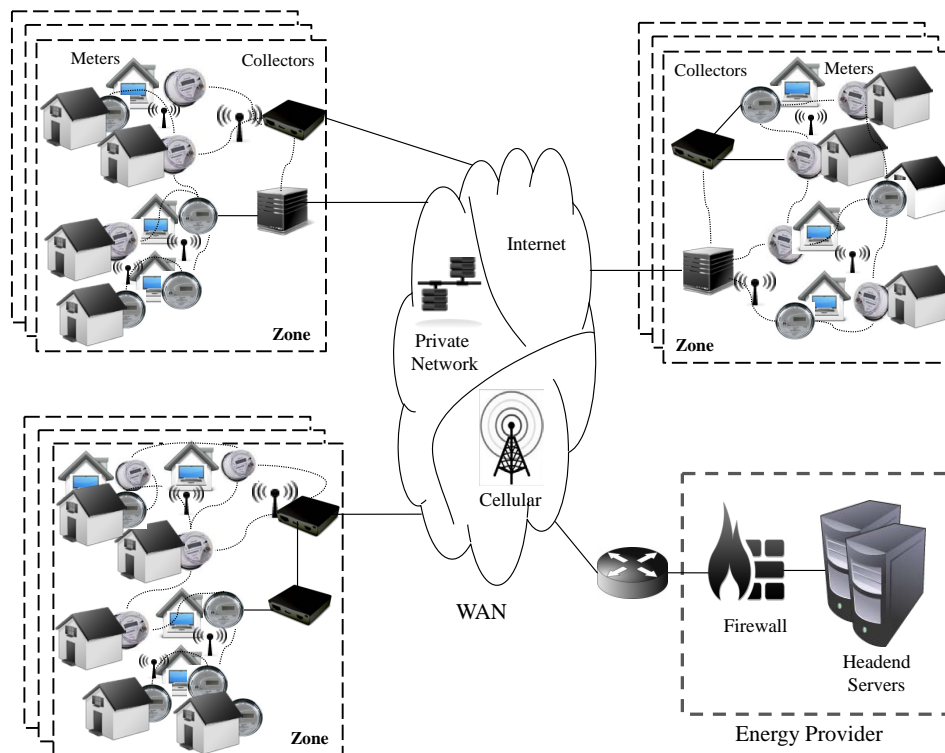
Figure 1. An example topology of an AMI, which often consists of many zones of smart meters where each zone usually represents meters located in geographically close one or more neighborhoods.

per 30 seconds, while each meter of another kind has a sampling rate of 20 KB per 40 seconds. Among these 100 meters, 60 of them fall into the first kind, while the rest of them fall into the second kind. Therefore, the collector will receive 3,120 KB (in an average) data in every 60 seconds, which is to be stored in its buffer. The size of the collector's buffer is 80 MB ($\approx$ 80,000 KB). Let us assume that the collector is configured with the report schedule, according to which the collector sends the data to the headend system in every 1440 seconds. Thus, in this reporting interval, 84,000 KB of total data will be reported to the collector by these meters. It is obvious that this amount of data (84,000 KB) will flood the collector's buffer (80,000 KB), which will in turn cause data loss (*i.e.*, initial 4,000 KB report data will be overwritten). Similarly, due to weak security controls or security policy misconfiguration, reachability or trusted communication can be failed [5].

Due to the lack of preparedness against failures, when an AMI device (*e.g.*, a collector) or a communication channel (*e.g.*, a link) fails, if there is no alternative paths to deliver or collect the data from the sender end to the receiver end, then the data delivery will be delayed or, even, lost. A loss or delay of data can easily disrupt different important utility services which require frequent real-time data [16]. For example, Demand Response Management Services (DRMS) typically reduce the overall cost of operating the electric system by real-time pricing as well as controlling the power generation efficiently by understanding the real-time demand from the real-time usage data [17]. The utility can provide efficient customer services like faster power restoration after outages using real-time data [18]. Thus, data loss or delayed data delivery will break these crucial services.

It is worth mentioning that the current practice of data collection from meters or collectors is occurred once or twice a day, in better cases hourly, while in worse cases monthly [19]. Moreover, at present DRMS are mostly not in operation as the projects of deploying smart meters are still continuing [14, 20]. In near future when the AMI system will be ideally established, the data collection rate will increase significantly, even per second. Then, the issue of data loss during data collection due to lack of resources (*e.g.*, data collectors), attacked or failure incidents, or improper data reporting scheduling will be very critical leading to disrupted services, decreased revenue, and loss of reputation.

2.3. Challenges and Objective

The correct functioning of an AMI system stands on dependable execution of tasks on time. The reliability of configuration depends not only on the local device parameters but also on the safe and secure interactions of these devices across the network. There is a significant number of logical constraints on configuration parameters of thousands of AMI devices, which need to be satisfied to ensure communications among AMI components, especially for the reliable data delivery from the meters to the headend system. These constraints represent system invariants as well as user-driven (*i.e.*, organizational) requirements. DHS AMI Task Force [11] and NIST [12] have provided guidelines toward resilient AMI systems. However, the challenge for ensuring the resiliency of an AMI system is to configure it in a correct manner considering the operational integrity and robustness properties. There is no such formal framework to support the energy providers for configuring an AMI system based on the essential and organizational resiliency requirements considering possible contingencies. Although a security verification framework has been proposed for AMI configurations in our previous work [5], the framework does not allow the automatic synthesis of security or resiliency configurations. This work focuses on satisfying the assured data delivery both in normal (no failure) and partially failed (link or device failures due to accidents or attacks) environments. This framework is easily extensible for further resiliency requirements. It is worth mentioning that our framework is easily extensible for further safety and robustness constraints beyond those mentioned in this paper.

2.4. Related Work

Throughout the last decade, the security policy misconfiguration and its verification were studied extensively, *e.g.*, in [21, 22]. In these approaches, formal definitions of configuration anomalies and safe deployment of security devices were proposed and algorithms were presented to discover configuration inconsistency. A good number of works have been done on risk-based security configuration analysis and security hardening, *e.g.*, [23, 24]. These works are not suitable for AMI as they do not consider the relations between the cyber and physical components and their required interactions.

A significant number of works (*e.g.*, [11, 12, 25]) have been initiated on describing the interoperability among heterogeneous smart grid components including security issues based on different attack scenarios. Researchers also extensively discussed the security and privacy challenges in AMI networks (*e.g.*, [2, 26, 27]). Wang et al. [28] presented an artificial intelligent-based approach for analyzing risks in smart grid

networks. Anwar et al. proposed a framework [29] for modeling power grid and its control elements using first order logic. McLaughlin et al. [30] described an approach for penetration testing on AMI systems. In our previous work [5], we presented a formal model based tool that provably verify operational consistency and security controls in AMI systems. Sgouras et al. [31] performed a qualitative assessment of the impact of cyber attacks on AMI. Gui et al. [32] demonstrated side-channel vulnerabilities on secure communication in smart metering infrastructures for software-based and hardware-based implementations. However, all these works specific to the smart grid security follow the traditional bottom-up approach of analyzing existing or deployed security policies.

The research on the security or resiliency architecture synthesis for cyber and cyber-physical systems is still in the early stage. Narain et al. presented a tool named ConfigAssure in [33], which takes routing specific security requirements and configuration variables as inputs and produces the values of the configuration variables as outputs that make the requirements true. Procedural approaches of generating firewall configurations presented in [34]. In another previous work [35–37], we proposed formal models for generating network security configurations satisfying the given isolation requirements and business constraints. Unlike to all of these works, in this paper, we solve the problem of synthesizing the smart grid's configurations for its operational integrity and robustness, focusing on AMI systems, where the resiliency requirements are significantly different than that of the traditional networks. Ghasempour and Moon [38] proposed a solution to find the optimum number of collectors for an AMI but it only considered minimizing the product of the collectors' cost and packet delay.

3. Synthesis Framework

Fig. 2 shows the architecture of the proposed synthesis framework which follows a top-down resiliency design automation approach. The framework includes the following major steps:

- Formal modeling of the AMI system, including the topology, devices, and data deliveries.
- Formal modeling of operational integrity and robustness requirements on top of the AMI system model, satisfaction of which determines AMI resiliency configurations, including a robust deployment of AMI collectors and their report schedules.
- Implementing or encoding of the model using SMT and solving it using an SMT solver.

The synthesis framework takes different inputs: (i) existing AMI topology including devices, (ii) resiliency design, and (iii) business requirements and constraints,
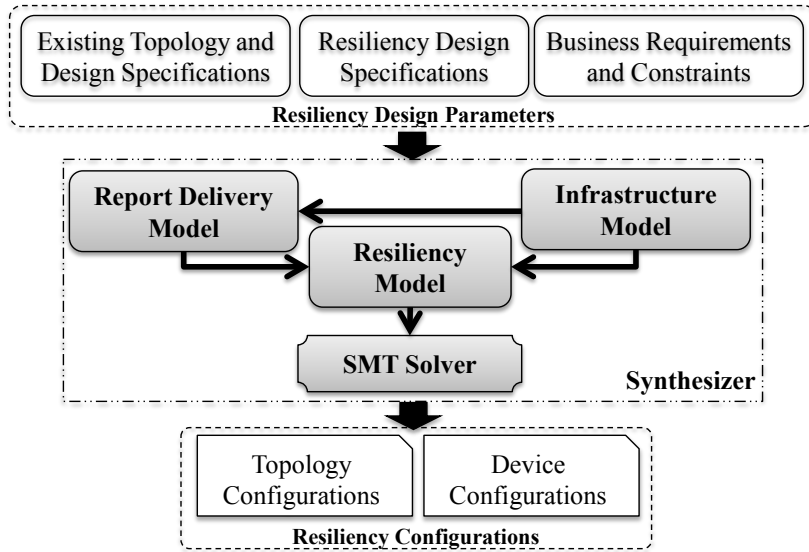
Figure 2. The architecture of the AMI resiliency configurations synthesis framework architecture.

including AMI topological and operational invariants, resiliency requirements, and the deployment budget. The resiliency requirements primarily include the maximum tolerable damage (*i.e.*, the loss of reported usage data) in the case of a contingency like failure of a collector or a communication path. In this work, for a particular AMI topology, smart meters are considered as already deployed, while the collectors are required to be deployed satisfying resiliency properties. We follow a group-based abstraction to formally model meters and collectors by considering similarities among their properties in order to scale with the large number of AMI devices. The solution to this model provides AMI topology configurations (*i.e.*, deployment design for necessary collectors) and AMI devices' configurations (*i.e.*, report schedules of the meters and the deployed collectors). The collector deployment plan includes where (resilient placements) and what (collector types and their numbers) to deploy.

4. Synthesis Model

We model the process of resiliency synthesis as a constraint satisfaction problem. In our notations, variables start with small alphabetic letters, while constants start with capital letters. We use multiple-letter notations to denote many parameters. We expect that these multiple-letter notations will help the readers to easily recall them. Also note that, no multiplication of two parameters is represented here without the multiplication sign.

Table 1. Important Modeling Parameters

Symbol	Type	Definition
k	Integer	Index to denote a zone.
i	Integer	Index to denote a meter group in a zone.
j	Integer	Index to denote a collector in a zone.
t	Integer	Index to denote a meter or collector type.
s	Integer	(Index to denote) a time slot.
$m_{k,i}$	Boolean	The i'th meter group is established in zone k.
$mT_{k,i}$	Integer	Type of the meters in group $m_{k,i}i$.
MT	Integer	Number of available meter types.
$mS_{k,i}$	Integer	Number of meters in group $m_{k,i}i$.
$MS_{k,t}$	Integer	Number of meters of type t in zone k.
$mS_{k,i,t}$	Integer	Number of meters of type t in group $m_{k,i}i$.
$mSR_{k,i}$	Integer	Sampling rate of each meter in group $m_{k,i}i$.
$mSS_{k,i}$	Integer	Sample size of each meter in group $m_{k,i}i$.
MSR_t	Integer	Sampling rate of a meter of type t.
MSS_t	Integer	Sampling size of a meter of type t.
$mRB_{k,i}$	Integer	Reporting base time of each meter in group $m_{k,i}i$.
$mRI_{k,i}$	Integer	Reporting interval of each meter in group $m_{k,i}i$.
$mC_{k,i}$	Integer	Primary collector associated with group $m_{k,i}i$.
$mC'_{k,i}$	Integer	Secondary collector associated with group $m_{k,i}i$.
$c_{k,j}$	Boolean	The j'th collector is deployed in zone k.
CN	Integer	Number of collectors maximally can be deployed in zone k.
$cT_{k,j}$	Integer	Type of collector $c_{k,i}i$.
CT	Integer	Number of available collector types.
$cBS_{k,j}$	Integer	Buffer size of collector $c_{k,i}i$.
$cC_{k,j}$	Integer	Deployment cost of collector $c_{k,i}i$.
CBS_t	Integer	Buffer size of a collector of type t.
CC_t	Integer	Deployment cost of a collector of type t.
$cRB_{k,j}$	Integer	Reporting base time of collector $C_{k,j}i$.
$cRI_{k,j}$	Integer	Reporting interval of collector $c_{k,j}i$.
$cIP_{k,j}$	Integer	Type of the path deployed for collector $c_{k,j}i$.
$PT_{k,j}$	Integer	Number of available path types.
$cPB_{k,j}$	Integer	Average bandwidth of the path deployed for collector $c_{k,j}i$.
$cPC_{k,j}$	Integer	The cost to deploy the path for collector $c_{k,j}i$.
$PB_{k,j}$	Integer	The bandwidth of the path of type t.
$PC_{k,j}$	Integer	The cost of the path of type t.
MBW	Integer	Bandwidth of the mesh network communication.
$MDMC$	Integer	Maximum allowable reporting latency from a meter to a collector.
MD	Integer	Maximum damage allowed in a contingency.
MC	Integer	Deployment budget.

4.1. AMI Configurations Parameters

We define various parameters, as shown in Table 1, to denote AMI (devices and topology) configurations.

Configuration Level Abstraction:

An enterprise AMI network typically consists of thousands of smart meters distributed over different geographical regions. The meters communicate with collectors for delivering data based on device configurations and communication properties. For the purpose of achieving better scalability, we apply the concept of abstraction in terms of groups or classes based on the similarities among the configurations of the meters. A particular group of meters shares the same (physical and logical) configuration properties. Collectors are modeled as individual devices. Moreover, we use the term *zone* to denote a collection of meters residing in a specific geographic location. The meters within a zone form a mesh network to communicate to collectors deployed in that zone. These meters also form one or more meter groups. Therefore, a meter group is identified or localized with respect to a zone.

AMI Device Configurations:

There are Z zones in the AMI system. There are one or more meter groups in a zone, while the number of groups in the zone is assumed to be no more than a threshold value (M). Meter group i ($1 \leq i \leq M$) in zone k ($1 \leq k \leq Z$) exists when $m_{k,i}$ is true. The synthesis framework is to populate properties of each existing meter group. Each group consists of a particular type ($mT_{k,i}$) of meters, and there are MT available types. Each group has a finite size, *i.e.*, the number of meters ($mS_{k,i}$). A particular type of meter is mainly specific to a vendor and it has a specific data sampling rate, *i.e.*, the number of samples per time slot ($mSR_{k,i}$), as well as a specific size for each sample ($mSS_{k,i}$). The report schedule is represented by two parameters, the base (starting) time of reporting ($mRB_{k,i}$) and the reporting interval ($mRI_{k,i}$), which indicate that the meters of this group report periodically to the collector associated with this group ($mC_{k,j}$) at each interval starting from the base time with respect to a specific time period (*e.g.*, a day). We assume minute as the unit for time slots and kilo bytes (KB) for the data or storage size.

Like a finite number of meter groups in a zone, we consider a finite number of (maximally) possible collectors in a zone. If $c_{k,j}$ is true, then collector j ($1 \leq j \leq C$) is deployed in zone k. The synthesis framework finds the collectors to be deployed in each zone. We model a collector's profile with two properties: the type of the collector ($cT_{k,j}$) and its report schedule. The type of the collector identifies the buffer size ($cBS_{k,j}$) and the deployment cost ($cC_{k,j}$). The report schedule of a collector usually represents the report requesting schedule of the headend system specific to the collector.

AMI Topology Configurations:

An AMI topology mainly defines the connectivity (*i.e.*, communication paths) between the AMI devices. As shown in Fig. 1, the AMI topology is well structured. The meters of a particular zone are considered to be connected to one or more collectors by forming a mesh network among them. We assume a similar kind of mesh network for each zone with an average bandwidth of MBW. A collector can be individually connected to the headend system through the WAN-based communication. A collector may be connected with another collector ($cFC_{k,j}$) to forward its stored report data to the headend. The individual path from a collector to the border router of the utility's network (*i.e.*, where the headend resides) can be wired, wireless, or cellular. The path selected for a collector ($cIP_{k,j}$) is associated with a bandwidth ($cPB_{k,j}$) and a deployment cost ($cPC_{k,j}$) with respect to the bandwidth (PB_t) and deployment cost (PC_t) its type (t). All of the collectors connected to the headend share the common path (with link bandwidth SBW) after the border router in the utility's network. The bandwidths of these communication paths play an important role for choosing the report schedules.

4.2. Modeling of AMI Physical Properties

AMI physical property modeling covers configurations for meters, collectors, and topology. The corresponding formalizations are presented in Table 2.

Meter Groups and their Properties:

The meters in a group have the same meter type. A valid meter type from MT available types is ensured in Equation 1. We assume that the meters are already deployed. That is, the number of a particular type of meters in a zone is given. Since a meter group in a zone has a specific type, the size of the group must be within the number of meters of that particular type residing in the zone (Equation 2). If the sizes of all meter groups in a zone having the same meter type are summed up, the result must be equal to the total of this particular type of meters in the zone (Equation 2). The sampling rate and the sample size of each meter of a meter group in a zone depend on its type. Equation 4 identifies the sampling rate and size of the meter with respect to those (MSR_t and MSS_t, respectively) of a meter of type t. The meters of a meter group in a zone send their sampled data to a specific collector (among CN collectors) deployed in the same zone. Finally, Equation 6 ensures that no two meter groups in a particular zone have the same values for all properties.

There is a finite number of collector types (CT) and a collector's type must be one of these types (Equation 8). The buffer size and deployment cost of each collector in a zone depend on its type as identified in Equation 9. If a collector is selected as the designated collector for

Table 2. Meter, Collector, and Topology Model

Meter Groups and their Properties:

$$m_{k,i} \rightarrow (mT_{k,i} \geq 1) \wedge (mT_{k,i} \leq MT) \quad (1)$$

$$m_{k,i} \rightarrow (mT_{k,i} = t) \rightarrow (mS_{k,i} \geq 1) \wedge (mS_{k,i} \leq MS_{k,t}) \quad (2)$$

$$\begin{aligned} m_{k,i} \wedge (mT_{k,i} = t) &\rightarrow (mS_{k,i,t} = mS_{k,i}) \\ \neg(m_{k,i} \wedge (mT_{k,i} = t)) &\rightarrow (mS_{k,i,t} = 0) \\ MS_{k,t} &= \textstyle\sum_i mS_{k,i,t} \end{aligned} \quad (3)$$

$$(mT_{k,i} = t) \rightarrow (mSR_{k,i} = MSR_t) \wedge (mSS_{k,i} = MSS_t) \quad (4)$$

$$m_{k,i} \rightarrow (mC_{k,i} \geq 1) \wedge (mC_{k,i} \leq CN) \quad (5)$$

$$\begin{aligned} m_{k,i} \wedge \quad m_{k,\hat{i}} &\wedge i \neq \hat{i} \rightarrow \\ \neg((mT_{k,i} = mT_{k,\hat{i}}) &\wedge (mC_{k,i} = mC_{k,\hat{i}}) \\ \wedge(mRB_{k,i} = mRB_{k,\hat{i}}) &\wedge (mRI_{k,i} = mRI_{k,\hat{i}})) \end{aligned} \quad (6)$$

$$\forall_{i>1} \; m_{k,i} \rightarrow m_{k,i-1} \quad (7)$$

Collectors and their Properties:

$$c_{k,j} \rightarrow (cT_{k,j} \geq 1) \wedge (cT_{k,j} \leq CT) \quad (8)$$

$$(cT_{k,j} = t) \rightarrow (cBS_{k,j} = CBS_t) \wedge (cC_{k,j} = CC_t) \quad (9)$$

$$\bigvee_i (mC_{k,i} = j) \rightarrow c_{k,j} \quad (10)$$

$$\forall_{i>1} \; c_{k,i} \rightarrow c_{k,i-1} \quad (11)$$

Topology and its Properties:

$$c_{k,j} \rightarrow (cIP_{k,j} \geq 0) \wedge (cIP_{k,j} \leq PT) \quad (12)$$

$$(cIP_{k,j} = t) \rightarrow (cPB_{k,j} = PB_t) \wedge (cPC_{k,j} = PC_t) \quad (13)$$

$$(cIP_{k,j} = 0) \rightarrow (cFC_{k,j} \geq 1) \wedge (cFC_{k,j} \leq CN) \quad (14)$$

$$(cFC_{k,j} = \hat{j}) \rightarrow \exists_{\hat{j}} \; c_{k,\hat{j}} \wedge (cIP_{k,\hat{j}} > 0) \quad (15)$$

a meter group in a zone for reporting, Equation 10 ensures that the particular collector is deployed.

Topology Properties:

A collector's individual communication path toward the headend system must be chosen from the available types (PT), while the type is zero when there is no path (Equation 12). Equation 13 associates the bandwidth ($cPB_{k,j}$) of the path and its deployment cost ($cPC_{k,j}$)

with those of the path type. If a collector does not have communication path to the headend system, it must be connected to a collector (in the same zone) to forward its report data to the headend (Equation 14). The forwarding collector should be a deployed one and it must have a communication path to the headend (Equation 15).

4.3. Modeling of AMI Resiliency Properties

AMI resiliency requirements are involved with operational integrity, data freshness, and robustness.

Report Schedule Constraints:

There are a finite number of potential values for the base time of the report schedule. If $\mathcal{B}_M$ and $\mathcal{B}_C$ are the set of potential base times for meters and collectors respectively, then $mRB_{k,i} \in \mathcal{B}_M$ and $cRB_{k,j} \in \mathcal{B}_C$. Similarly, there is a finite set of potential values for reporting intervals. There are invariants that must be followed to select report schedules: (ii) the base time of a report schedule is lower than its interval to guarantee full cycle, and (ii) a collector reports less frequently than its associated meters to ensure fresh data reporting to the headend. Equation 16 specifies these constraints:

$$\begin{aligned} m_{k,i} &\rightarrow mRB_{k,i} < mRI_{k,i} \\ c_{k,j} &\rightarrow cRB_{k,j} < cRI_{k,j} \\ (mC_{k,i} = j) &\rightarrow mRI_{k,i} \leq cRI_{k,j} \end{aligned} \quad (16)$$

The total incoming data from the meters within the report interval of the collector should not exceed its buffer. Moreover, the reporting data should not exceed the communication bandwidth. Let $mRA_{k,i,s}$ denote if a meter reports at a particular time slot (s) and $mRS_{k,i}$ specify the size of the reported data. Similarly, there are $cRA_{k,j,s}$ and $cRS_{k,j}$ for collectors. We calculate $mRA_{k,i,s}$ (and similarly $cRA_{k,j,s}$) as follows:

$$mRA_{k,i,s} \rightarrow m_{k,i} \rightarrow ((s - mRB_{k,i})\% mRI_{k,i} = 0) \quad (17)$$

We compute the report size of a meter group considering the average number of times each meter sends data to the associated collector during the reporting interval. The same is calculated for each collector. That is:

$$m_{k,i} \rightarrow (mRS_{k,i} = mS_{k,i} \times mSS_{k,i} \times mRI_{k,i}/mSR_{k,i}) \quad (18)$$

$$c_{k,j} \rightarrow (cRS_{k,j} = \sum_{i|(mC_{k,i}=j)} mRS_{k,i} \times cRI_{k,j}/mRI_{k,i}) \quad (19)$$

With the above calculation of $cRS_{k,j}$, Equation 20 ensures no overwrite on the stored data in the buffer.

$$c_{k,j} \rightarrow cRS_{k,j} \leq cBS_{k,j} \quad (20)$$

The meters and collectors in a zone share a data transmission bandwidth (*i.e.*, having shared data

throughput) of the mesh network. Therefore, to ensure the successful report delivery to a collector from the associated meters, the accumulated rate of data transmission by the meters must be within the bandwidth:

$$\sum_{i|m_{k,i}} mRS_{k,i} \leq MBW \times mRI_{k,i} \qquad (21)$$

If collector j in zone k is not connected to the border router of the utility's network directly, it sends the data to a neighboring collector ($cFC_{k,j}$) according to its own schedule. We assume that the communication latency between neighboring collectors are negligible compared to the long distance toward the headend. Thus, for each time slot (s), the communication bandwidth constraint for the communication from collector j to the border router is formulated considering the reporting schedules of itself and the collectors forwarding to it, and the associated report sizes:

$$cRA_{k,j,s} \times cRS_{k,j} + \sum_{\hat{j}|(cFC_{k,j}=j)} (cRA_{k,\hat{j},s} \times cRS_{k,\hat{j}})$$
$$\leq cPB_{k,j} \times cRI_{k,j} \qquad (22)$$

Since the path after the border router is shared by all of the collectors, a bandwidth constraint is also considered by summing up all the reports at a particular time slot.

The grid operators may have constraints on the quality of the data delivery, especially with respect to the reporting delay. For example, a meter should report its usage data to the associated collector within a particular time interval, while a collector should not delay in forwarding the data to the headend system more than a threshold time. These constraints are reflected in choosing $\mathcal{B}_\mathcal{M}$ and $\mathcal{B}_\mathcal{C}$. There can be constraints to limit the data transmission delay. For example, the data transmission delay from a meter to a collector should reach within a threshold time ($MDMC$). In the case of simultaneous reporting, *i.e.*, data delivery at the same time slot, the total data should be delivered within the threshold time. Therefore:

$$\sum_{i|m_{k,i} \wedge mRA_{k,i,s}} mRS_{k,i} \leq MBW \times MDMC \qquad (23)$$

A similar constraint on the data transmission delay can be applied for collectors to the headend system.

Robustness Constraints:

The robustness or fault-tolerance policy often states that when one or more intermediate devices (*i.e.*, collectors) or one or more communicating links fail, the system still can operate without any damage (*e.g.*, data loss). Power grids often consider $n - 1$ contingency-constrained policy, especially for transmission systems,

which ensures a single node (*i.e.*, a load, generator, or transmission line) fault-tolerance. We assume a zone-based fault-tolerance requirement, which is limited to failures of collectors and the communication paths from the collectors to the headend/utility network. It is worth mentioning that, as Mesh networks of smart meters are considered for the connectivity among meters and collectors, these networks provides alternative paths in the cases of intermediate meter or link failures toward the collector. The secure and robust deployment of a mesh network has been addressed in the literature (*e.g.*, [39] and we consider this deployment out of the scope of this work. Therefore, we focus on the AMI network's resiliency by making each zone robust against a single collector/collector-utility communication path failure. Therefore, if there are Z zones in an AMI system, then we will achieve reliability against a $Z(n - 1)$ contingency, *i.e.*, a Z-node fault-tolerance. We define robustness as one minus damage, where the damage is the number of meters whose usage data is not ensured to be delivered to the headend system in the case of a failure.

A system is robust when there are alternatives to perform necessary operations. These alternatives can be found only if there is necessary redundancy. For example, if one collector is sufficient for the meters in a zone, another collector is required for a single collector failure. Moreover, there should have alternative paths toward the headend. Since the meters form a mesh network in a zone, and the collectors are connected to this mesh network, a meter has a robust communication to different collectors through the same network.

There are a number of constraints which must be satisfied for resiliency. If a collector fails, the rest of the collectors in a zone must have enough buffer space to store the data reported by the meters of the zone. Since different collectors often have different report schedules, we can describe the same constraint in different words: the total data reported by the meters during a cycle period (P, *e.g.*, a day) must be less than or equal to the maximum possible data that the rest of the collectors can store (*i.e.*, total buffer sizes) throughout the period without any overwrite. The following equation formalizes this constraint for each collector's failure:

$$rC_{k,j} \rightarrow \sum_{j|c_{k,j}} cRS_{k,j} \times P/cRI_{k,j}$$
$$\leq \sum_{j|(j \neq \hat{j}) \wedge c_{k,j}} cBS_{k,j} \times P/cRI_{k,j} \qquad (24)$$

Although communication paths can be established between collectors of neighboring zones to cover-up a link failure, we consider redundant paths through collectors deployed in a zone. The collectors will provide alternative paths to the headend, as the

collectors in a zone are connected to each other (often through the mesh access points), while multiple collectors must be connected to the utility (up to the border router) through separate communication paths. The following equation ensures enough bandwidth to forward the stored usage data to the headend system in the case of a link failure:

$$rC_{k,j} \to \sum_{j|c_{k,j}} cRS_{k,j}/cRI_{k,j} \\ \leq \sum_{\hat{j}|(j \neq \hat{j}) \wedge c_{k,\hat{j}} \wedge (cIP_{k,\hat{j}} > 0)} cPB_{k,\hat{j}} \tag{25}$$

A meter group is robust with respect to its collector failure or the associated communication path failure:

$$rM_{k,i} \to m_{k,i} \wedge \exists_j \left((mC_{k,i} = j) \wedge rC_{k,j} \right) \tag{26}$$

Finally, the robustness constraint that specifies the maximum possible damage (*MD*), *i.e.*, the maximum number of meters whose data may not reach the headend system is formalized as follows:

$$\sum_{i|\neg rM_{k,i}} mRS_{k,i} \leq MD \tag{27}$$

Deployment Cost Constraint:

Since the grid operator has a limited budget, the total cost for deploying collectors and communication paths (from collectors to the utility's network). If *MC* is budget, this constraint is formalized as follows:

$$\sum_{\{k,j\}|c_{k,j}} cC_{k,j} + \sum_{\{k,j\}|c_{k,j} \wedge (cIP_{k,j} > 0)} cPC_{k,j} \leq MC \tag{28}$$

4.4. An Example Case Study

We present an example case study illustrating the execution of the proposed synthesis framework.

Implementation:

We encode the system configuration and the constraints into SMT [3]. We encode our formalizations mainly using Boolean (for logical variables like $m_{k,i}$s, $c_{k,j}$s, etc.) and integer (for property variables like $mSS_{k,i}$s, $cBS_{k,j}$s, etc.) terms. We use real terms for some variables (*e.g.*, $mRS_{k,i}$s and $cRS_{k,j}$s), where either they can take real values or they are used in division generating fractions. The execution of the model (in Z3) provides either satisfiable (*sat*) or unsatisfiable (*i.e.*, no solution exists) result. If it is *sat*, the necessary AMI configurations are found from corresponding variables.

An Example:

In this example, we consider an arbitrary AMI system of 1,000 smart meters distributed in 4 zones. The input of the example is shown in Table 3. There are 2 types

Table 3. Input to the Example (Partial)

```
# Number of meters and Zones
1000 4
# Distribution of meters in each zone based on meter types
350 1 150 2 200
120 1 60 2 60
230 1 100 2 130
300 1 200 2 100
# Number of meter types
2
# Meter sampling properties (interval in minute, size in KB)
5 2
10 3
# Max number of meter groups in a zone and min number of
# meters in a group
15 20
# Number of collector types
2
# Collector properties (buffer size (KB) and deployment cost (k$))
10000 6
12000 10
# Maximum number of collectors per zone
8
# Potential paths from a collector to the utility's network
3
25 50 100 # Bandwidth (kbps)
15 18 25 # Cost ($k)
# Shared path bandwidth (meter to collector, collector to headend)
100 1200
# Max reporting delay (meter to collector, collector to headend)
# and % of data that must satisfy this freshness constraint
10 30 80
# Max reporting delay from collector to headend in contingency
# and % of data that must satisfy this freshness constraint
40 70
# Max data loss in contingency
10
# Budget (cost constraint in k$)
250
```

of meters and the number of each type of meters in a specific zone is shown. A type 1 meter takes a sample (of size 2 KB) at each 5 minutes, while a type 2 meter takes a sample (of size 3 KB) at each 10 minutes. A collector can be either of 2 types, while each type has different buffer size (*e.g.*, type 1 has 10,000 KB buffer, while type 2 has 12,000 KB buffer) and deployment cost.

The minimum reporting interval for a meter is considered as 30 minutes, while the maximum is 120 minutes, while they are 120 minutes and 360 minutes for collectors. The maximum number of meter groups expected in a zone is 15, while each group should have at least 20 meters. The maximum number of collectors that can be deployed in a zone is 8. The communication bandwidth between meters and collectors (*i.e.*, the mesh network) is 100 kbps. The individual link from a collector to the utility's border router can be of 3 types with different bandwidths and deployment costs. The shared bandwidth from the utility border router toward the headend system has 1200 kbps. The organizational requirements specify the data freshness and robustness constraints. According to the freshness constraint, at least 80% of the data should reach (*i.e.*, the transmission

Table 4. Output: Meters' Configurations

| Zone	Group Id	Meter Type	Group Size	Pri. Col.	Sec. Col.	R. Base Time	R. Interval
1	1	2	13	7	6	30	60
1	2	1	2	7	3	60	120
1	4	1	2	7	3	10	120
1	6	1	2	7	3	30	60
1	8	1	5	6	3	0	30
1	9	1	2	7	3	0	60
1	10	1	2	6	3	30	60
1	12	2	2	3	7	60	120
1	13	2	2	6	3	0	120
1	14	2	3	3	7	0	30
2	3	1	2	6	8	0	60
2	6	2	2	6	8	0	120
…	…	…	…	…	…	…	…

Table 5. Output: Collectors' Configurations

Zone	Col. Index	Col. Type	R. Base Time	R. Interval	Path Type	Pri. Forw. Coll.	Sec. Forw. Coll.
1	3	1	60	120	2	-	-
1	6	2	60	120	3	-	-
1	7	2	60	120	-	3	6
2	6	1	0	120	1	-	-
2	8	2	60	120	1	-	-
3	4	2	60	120	3	-	-
…	…	…	…	…	…	…	…

delay) from a meter to a collector in 10 minutes, while from a collector to the headend system in 30 minutes. The robustness constraint specifies that the maximum data loss in a contingency (*i.e.*, in the case of a single node or link failure) is no more than 10%. The collector deployment budget is $250K.

Our formal model corresponding to this example returns a satisfiable result along with the synthesis of necessary configuration parameters. The configurations associated with the meters and the deployed collectors are shown (partially) in Tables 4 and 5. We observe that 10, 6, 2, and 14 meters groups are selected for zones 1, 2, 3, and 4, respectively. The primary and secondary collectors associated to each meter group are identified. In zone 1, for example, collector 7 is associated with meter group 4 as the primary collector, while collector 3 is selected as the secondary (when the primary is unavailable).

With regards to the collector deployment, 3 collectors are selected to be deployed in zones 1 and 4, while 2 collectors are selected for each of the remaining zones. Collector 7 in zone 1 and collector 5 in zone 4 do not have direct communication paths to the headend. Therefore, primary and secondary collectors are selected for them for data forwarding. For example, collector 7 in zone 1 has collectors 3 and 6 as primary and secondary forwarding nodes, respectively. The report schedules are selected in such a way that the collectors do the reporting in distributed time slots, which also consider the limited shared bandwidth.


(a)

(b)
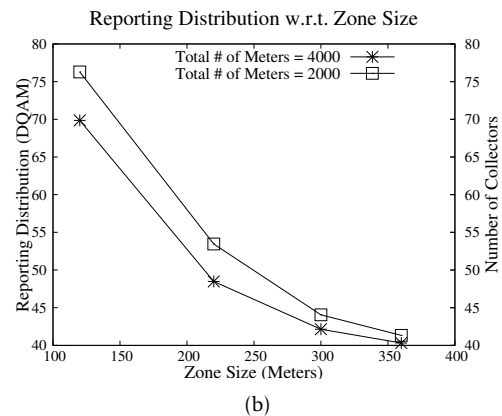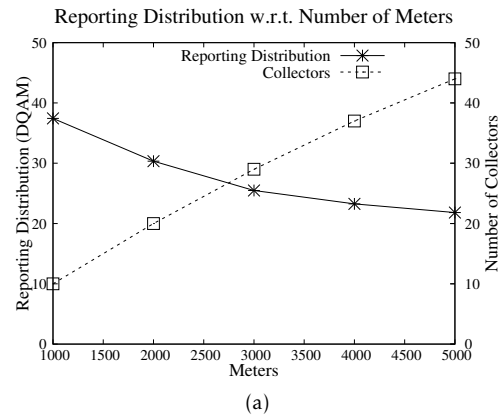
Figure 3. The impact of number of meters on the reporting schedule at (a) the collector level and (b) the meter level.

5. Evaluation

We first perform the performance analysis of our proposed framework in discerning the impact of different factors in resilient AMI design. We then extensively evaluate the scalability of the tool.

5.1. Methodology

We evaluate the proposed synthesis model in terms of different metrics primarily by varying the AMI network size. We consider the network size as the total number of smart meters, which are distributed in different sizes of zones. We consider only a single headend system in the network. We vary the number of smart meters from 1000 to 10000 and while the size of each zone is considered between 200 and 500 meters. Since an organization usually is limited within the choice of a few types of meters and collectors, we consider up to 3 types of meters or collectors in our experiments. The deployment cost of a particular type of collector is taken arbitrarily. The number of potential values of the reporting base time as well as the interval is kept less
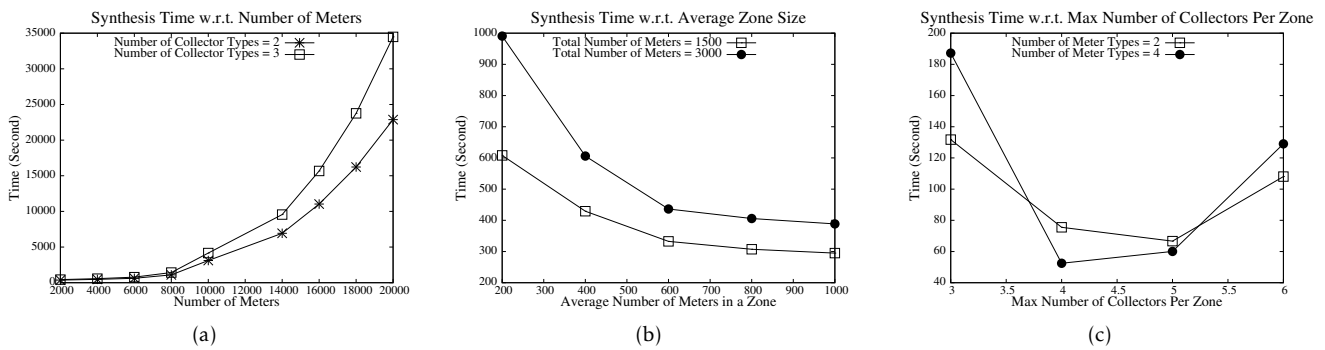
Figure 4. The execution of the synthesis framework with respect to (a) the number of meters, (b) the average size of each zone, and (c) the maximum number of collectors per zone.

than or equal to 10. We run our experiments on an Intel Core i5 machine with 16 GB memory.

5.2. AMI Configuration Analysis

Various factors, such as available resources (*e.g.*, data buffer, bandwidth, budget, etc.) and network size, impact the AMI configuration (*e.g.*, the deployment of collectors and their reporting schedules). For example, in the case of the shared path from the collectors to the headend, if the number meters (and so the collectors) reporting to the headend increases, the collectors' reporting schedules require to be less overlapping (*i.e.*, reporting load distributed more uniformly). To analyze this feature, we consider the difference between the quadratic and arithmetic means (DQAM) of the reporting size over the potential reporting time slots. As we know, the arithmetic mean will be equal to less than the quadratic mean, and the difference becomes larger when the reporting size is less uniformly distributed. The corresponding evaluation results presented in Fig. 3(a) reflects this conjecture. In this simulation, we vary the AMI network size (*i.e.*, the number of total meters) and increase the deployment cost proportionally, but the shared path's bandwidth remains the same. As the figure shows, with the network size, the number of collectors increases, while the corresponding DQAM decreases. We observe a similar behavior in Fig. 3(b) in which we consider different zone sizes.

5.3. Scalability Analysis

The scalability of the proposed tool is evaluated in terms of its execution time and memory usage.

Impact of the Problem Size on Execution Time. Fig. 4(a) shows the execution time of our synthesis framework with respect to the AMI size, *i.e.*, the number of smart meters. We show the execution time in two

different scenarios of the number of collector types. The graphs in the figure show that with the number of meters, the increase in the execution time lies between linear and quadratic growths. Although the number of parameters seems to be increased exponentially (as does the execution time), we observe complexity less than that. This is due to the application of the property-based abstraction (*i.e.*, the grouping of the meters when they share the same properties). It is to be noted that the execution time primarily depends on the number of clauses and the operational complexity. The order of the clause number is cubic for each zone, which depends on the numbers of meter groups, collectors, and time slots (*e.g.*, Equation 17). The complexity of the clauses has the maximum cubic order (*e.g.*, Equation 28). The abstraction of the meters into groups and types helps improve the ultimate performance.

The evaluation results with respect to the average size of each zone are shown in Fig. 4(b). The graphs shows that the execution time decreases with the zone size. If the size increases, the number of zones reduces in the AMI network, which ultimately reduces the effective problem size. We also evaluate the execution time by changing the maximum number of collectors in each zone and we observe that, as shown in Fig. 4(c), the time decrease with the increase of the number of possible collectors, while starts to increase after some point. Although the increase in the maximum number of collectors increases the solution space (thus reduces the execution time), more increase in the number leads to unnecessary selections of larger numbers of collectors beyond the budget.

Impact of the Constraints on Execution Time. The synthesis of AMI configurations depends on the given constraints, *e.g.*, the budget, freshness, and data loss requirements. However, the tighter the constraint, the more time is required to synthesize the configurations. We analyze the impact of this budget (*i.e.*, the deployment cost limit), on the execution time. The analysis results are
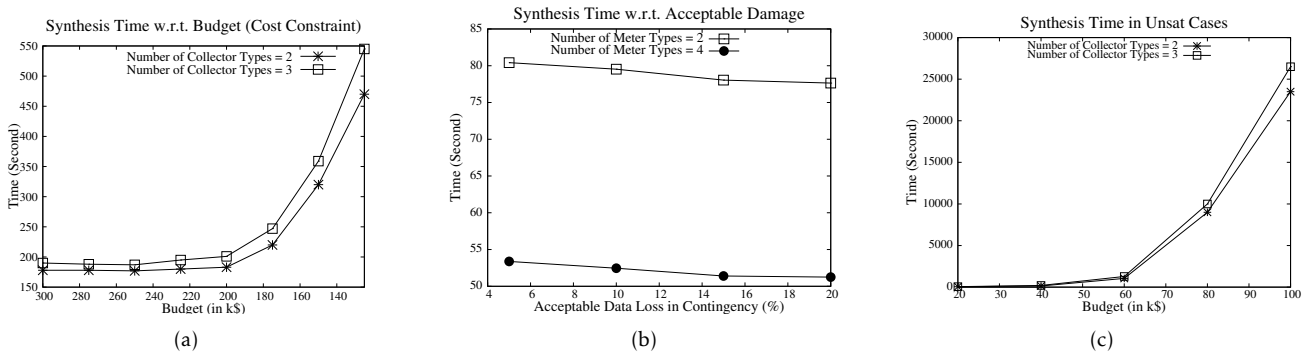
Figure 5. The execution time with respect to (a) deployment budgets, (b) data loss thresholds, and (c) unsatisfiable cases.

Table 6. Memory Requirements (in MB)

Hosts	Scenario 1	Scenario 2
1000	45.20	48.60
2000	109.60	122.70
3000	185.30	193.30
4000	366.50	375.80
5000	514.40	528.90

shown in Fig. 5(a). The graphs show that the execution time increases exponentially with the decrease of the budget. This is because the lower the budget, the more search is required by the solver to find a satisfiable set of configurations, and thus the execution time increases. When the budget becomes very low, the time grows rapidly. After a certain point, there will be no solution (an unsatisfiable case) due to insufficient budget.

We analyze the impact of the data loss constraint on the execution time and we observe that the execution time reduces with the decrease of the requirement, *i.e.*, the increase in the acceptable data loss (Fig. 5(b)). However, if these constraints become tighter, *e.g.*, the budget is low, there may be no solution. In unsatisfiable cases, the execution time is often high, as whole of the search space needs to be traversed to conclude that there is no solution. However, if a constraint is too tight (*e.g.*, the budget is too low), the solver takes a much shorter time to conclude with unsatisfiability. In such cases, the potential space is small due to the highly tight constraints. Fig. 5(c) shows the evaluation results in unsatisfiable cases.

Memory Requirement. We evaluate the memory requirement for executing our model in the SMT solver [3] by changing the number of meters. The memory requirement mainly includes the memory required for the variables that we use in modeling, and the intermediate variables that the solver uses to implement the theories applied in our constraint modeling. The analysis results are shown in Table 6 for two different scenarios. In the first and second scenarios, the number of collector types is 2 and 3, respectively. We observe that

the memory requirement lies between the linear and quadratic orders with respect to the number of meters. The table shows that the memory requirement in the second scenario is larger than the memory requirement in the first because, due to a larger number of collector types, there are more options (and so more variables) to design the deployment of collectors.

6. Discussion

In this section, we discuss the issues about the usability of the proposed AMI configuration synthesis framework.

6.1. Scalability and Abstraction

AMISynth performs automatic synthesis of the AMI configuration that mitigates potential threats by eliminating misconfigurations and keeps the damage limited under contingencies by introducing robustness. It shows high efficiency even for a network with thousands of smart meters. However, to achieve this scalability, the tool embraces a couple of limitations. First, it uses device and property level abstraction, especially to deal with large-scale smart grid configuration, which may not provide fine-grain resiliency measures. Second, the tool uses an SMT solver as its core analysis engine that requires different normalizations for efficient real-valued calculations, reducing accuracy.

6.2. Resiliency Specifications

Although AMISynth provides resiliency against Z failures maximally for an AMI, each zone tolerates no more than a single fault. In a zone that refers to a geographical location, we often see one or a very few data collectors, which accumulate data from many smart meters because the deployment plan usually ignores robustness/resiliency in practice. Since we assume failures concerning collectors and communication paths from collectors to the utility

network, a single node contingency is enough to demonstrate the practicality. There can be failures of multiple smart meters. However, since we assume a mesh network connecting smart meters and collectors in a zone, failures of meters or communication paths are dealt with by the mesh topology/protocol by providing alternative routes in the cases of intermediate meter/link failures toward the collector. It is worth mentioning that we divide the AMI into different Zones for the tractability of formal modeling, where the smart meters are geographically co-located and has the same data concentration points.

7. Conclusion

Automated synthesis of AMI resiliency configurations is an important and challenging problem. In this paper, we address this challenge by presenting an automated AMI configuration synthesis framework. We model various constraints that are crucial for resilient data delivery in AMI systems. We model the resiliency by introducing redundancy in the AMI design, which is extendable for further resiliency requirements. The execution of the proposed model synthesizes necessary resiliency configurations satisfying the constraints. We implement the framework using SMT and evaluate its scalability in different synthetic AMI networks and requirements. We achieve significantly high scalability by applying the group-based abstraction in the model. The evaluation results shows that to synthesize the resiliency configurations for a AMI network of 10,000 meters takes less than an hour in our particular computing environment. In the future, we would like to address the resiliency architecture design problem for Supervisory Control and Data Acquisition (SCADA) networks.

References

[1] Punya Prakash. Data concentrators: The core of energy and data management. http://www.ti.com/lit/wp/spry248/spry248.pdf?DCMP=induvid1&HQS=ep-pro-sit-induvid1-toolsinsider-20140605-mc-en, 2013. Texas Instruments.

[2] P. McDaniel and S. McLaughlin. Security and privacy challenges in the smart grid. *IEEE Security and Privacy*, 7(3):75–77, June 2009.

[3] L. Moura and N. Bjørner. Formal methods: Foundations and applications. chapter Satisfiability Modulo Theories: An Appetizer, pages 23–36. Springer-Verlag, 2009.

[4] M. A. Rahman and E. Al-Shaer. Formal synthesis of dependable configurations for advanced metering infrastructures. In *SmartGridComm*, pages 289–294, November 2015.

[5] M. A. Rahman, E. Al-Shaer, and P. Bera. Smartanalyzer: A noninvasive security threat analyzer for ami smart grid. In *31st IEEE INFOCOM*, pages 2255–2263, March 2012.

[6] Ramyar Mohassel, A. Fung, F. Mohammadi, and K. Raahemifar. A survey on advanced metering infrastructure. *Intl. Journal of Electrical Power and Energy Systems*, 63:473 – 484, 2014.

[7] B. Karimi, V. Namboodiri, and M. Jadliwala. On the scalable collection of metering data in smart grids through message concatenation. In *IEEE Intl. Conf. on Smart Grid Communications*, pages 318–323, Oct 2013.

[8] The EEI and AEIC Meter Committees. Smart meters and smart meter systems: A metering industry perspective. http://www.eei.org/issuesandpolicy/grid-enhancements/Documents/smartmeters.pdf.

[9] Lontalk protocol specification. http://www.enerlon.com/JobAids/Lontalk%20Protocol%20Spec.pdf.

[10] J. Wenger and B.C. Le. System and method to manage utility meter communications, 2014.

[11] AMI-SEC Task Force. Ami system security requirements, 2008. http://osgug.ucaiug.org/utilisec/amisec/.

[12] Smart Grid Interoperability Panel-Cyber Security Working Group. Nistir 7628: Guidelines for smart grid cyber security, 2010.

[13] R. Alimi, Y. Wang, and Y. R. Yang. Shadow configuration as a network management primitive. In *ACM SIGCOMM Conference on Data communication*, pages 111–122, 2008.

[14] HP Technical. Using the hp vertica analytics platform to manage massive volumes of smart meter data. http://www.vertica.com/wp-content/uploads/2014/05/SmartMetering_WP.pdf.

[15] IBM Software. Managing big data for smart grids and smart meters. http://www.smartgridnews.com/artman/publish/Business_Strategy/Managing-big-data-for-smart-grids-and-smart-meters-5248.html.

[16] Honeywell. The communications requirements of electric utilities. http://energy.gov/gc/downloads/nbp-rfi-communications-requirements-honeywell-responses-request-information-rfi.

[17] J.S. Vardakas, N. Zorba, and C.V. Verikoukis. A survey on demand response programs in smart grids: Pricing methods and optimization algorithms. *IEEE Communications Surveys Tutorials*, pages 1–1, 2014.

[18] Silver Spring. How the smart grid makes restoration faster and easier for utilities. http://www.silverspringnet.com/outage/pdfs/SilverSpring-Whitepaper-Outage.pdf.

[19] UK Energy Supplier and Energy Company-E.ON. Smart meter frequently asked questions. https://www.eonenergy.com/for-your-home/saving-energy/smart-meters/frequently-asked-questions.

[20] Energy Research Council. Best practices: Demand response. http://energyresearchcouncil.com/best-practices-demand-response.html.

[21] X. Ou, S. Govindavajhala, and A. Appel. Mulval: A logic-based network security analyzer. In *14th USENIX Security*, pages 113–128, July 2005.

[22] E. Al-Shaer, W. Marrero, A. El-Atawy, and K. Elbadawi. Network configuration in a box: towards end-to-end verification of network reachability and security. In *IEEE ICNP*, pages 107–116, 2009.

[23] R. Dewri, N. Poolsappsi, I. Ray, and D. Whitley. Optimal security hardening using multi-objective optimization

on attack tree models of networks. In *14th ACM CCS*, pages 204–213, 2007.

[24] J. Homer and X. Ou. Sat-solving approaches to context-aware enterprise network security management. *IEEE JSAC*, 27(3):315–322, 2009.

[25] I. Poursanidis, E. Kotsakis, N. Andreadou, and M. Masera. Evaluation of interoperability in the context of advanced metering infrastructure. In *IEEE International Conference on Smart Energy Grid Engineering (SEGE)*, pages 121–127, 2018.

[26] S. McLaughlin, D. Podkuiko, and P. McDaniel. Energy theft in the advanced metering infrastructure. In *CRITIS*, pages 176–187, 2009.

[27] U.S. Department of Energy. Advanced metering infrastructure and customer systems. https://www.energy.gov/sites/prod/files/2016/12/f34/AMI%20Summary%20Report_09-26-16.pdf.

[28] Y. Wang, D. Ruan, J. Xu, M. Wen, and L. Deng. Computational intelligence algorithms analysis for smart grid cyber security. *Advances in Swarm Intelligence*, 6146:77–84, 2010.

[29] Z. Anwar, R. Shankesi, and R. Campbell. Automatic security assessment of critical cyber-infrastructures. In *IEEE/IFIP DSN*, 2008.

[30] S. McLaughlin, D. Podkuiko, S. Miadzvezhanka, A. Delozier, and P. McDaniel. Multi-vendor penetration testing in the advanced metering infrastructure. In *ACSAC*, pages 107–116, 2010.

[31] K. I. Sgouras, A. D. Birda, and D. P. Labridis. Cyber attack impact on critical smart grid infrastructures. In *ISGT 2014*, pages 1–5, 2014.

[32] Y. Gui, A. S. Siddiqui, S. M. Tamore, and F. Saqib. Security vulnerabilities of smart meters in smart grid. In *IECON 2019 - 45th Annual Conference of the IEEE Industrial Electronics Society*, volume 1, pages 3018–3023, 2019.

[33] S. Narain, G Levin, S Malik, and V. Kaul. Declarative infrastructure configuration synthesis and debugging. *JNSM*, 16(3):235–258, 2008.

[34] B. Zhang and E. Al-Shaer. Synthesizing distributed firewall configurations considering risk, usability and cost constraints. In *CNSM*, 2011.

[35] M.A. Rahman and E. Al-Shaer. A formal framework for network security design synthesis. In *IEEE 33rd International Conference on Distributed Computing Systems (ICDCS)*, pages 560–570, July 2013.

[36] M.A. Rahman and E. Al-Shaer. A formal approach for network security management based on qualitative risk analysis. In *IFIP/IEEE International Symposium on Integrated Network Management (IM)*, pages 244–251, May 2013.

[37] M. Rahman and E. Al-Shaer. Automated synthesis of distributed network access controls: A formal framework with refinement. *IEEE Transactions on Parallel and Distributed Systems*, 28(2):416–430, 2017.

[38] A. Ghasempour and T. K. Moon. Optimizing the number of collectors in machine-to-machine advanced metering infrastructure architecture for internet of things-based smart grid. In *IEEE Green Technologies Conference (GreenTech)*, pages 51–55, 2016.

[39] Jing Dong. *Secure and Robust Communication in Wireless Mesh Networks*. PhD thesis, Purdue University, 2009.