Robust and Real-time Communication on Heterogeneous Networks for Smart Distribution Grid

Kubilay Demir, Daniel Germanus and Neeraj Suri
{kubidem, germanus, suri}@cs.tu-darmstadt.de, TU Darmstadt, Germany

***Abstract*—In the power distribution domain, the increasing penetration of Distributed Generation (DG) requires reliable, real-time and cost effective communication in order to manage the grid safely and securely. To achieve this goal, a heterogeneous network (mixing the public Internet and private networks) is a promising solution due to its cost effectiveness. However, the current Internet infrastructure does not support reliable real time communication. To cope with these challenges, we propose QoS routing based on a novel overlay network protocol. Moreover, the protocol provides fault-tolerant communication for critical applications by applying multi-path routing over disjoint paths. Simulation results demonstrates that the proposed overlay network and algorithms perform well in obtaining QoS-aware overlay routing service in scalable manner as well as fault-tolerance for the critical application.**

***Keywords*- Smart Distribution Grid, Overlay network, QoS Routing, Fault-tolerant.**

I. INTRODUCTION

The advent of electric storage and distributed generation (DG) requires active load balancing plus handling variable and intermittent generation sources, within a power distribution system that traditionally only has one-way flows of electricity. To cope with these challenges, many Distribution Automation (DA) applications recently have been proposed . These applications impose stringent communication requirements (regarding reliability, robustness, timeliness and QoS etc.), particularly for Islanding Protection based on communication [5], which permits DG to be tripped in order to avoid damage to the distribution grid.

To satisfy these communication requirements, utilities target dedicated private end-to-end (E2E) the networks. However, this may not always be achievable due to budget and technical feasibility constraints. Realistically, the overall network architecture comprises heterogeneous private and public networks, which potentially consist of multiple Autonomous Systems (AS) [8]. Unfortunately, the current Internet infrastructure does not inherently provide the necessary service guarantees for such safety-critical applications which require both low latency and high reliability. One reason for this deficiency is the Border Gateway Protocol (BGP) routing policy and long re-convergence time. Moreover, in such heterogeneous networks, E2E QoS cannot be guaranteed by employing the current underlying QoS approaches (e.g., Diffserv, IntServ). As their data prioritization mechanisms do not match DA applications' requirements. For instance, in the DA applications lower latency does not always imply higher priority, but in Diffserv, low latency means low priority [6].

There have been efforts to meet the stringent requirements of Smart Grid (SG) applications, e.g., the INTEGRIS project [9] mainly focuses on QoS Routing by employing QoS Broker device in heterogeneous OSI layer 2 technologies. Furthermore, GridStat [10] proposes a pub-sub network of message routers controlled by QoS brokers to satisfy the NASPInet QoS requirements. However, INTEGRIS does not address the public network but only dedicated networks. GridStat assumes that the network topology is static and already known and needs certain QoS guarantees from the underlay network.

In addition, to enhance real-time applications' performance on the Internet, many studies have been proposed [11], [12]. These works employ path diversity to enhance the performance of the applications when the default paths cannot meet their requirements. They take advantage of intermediate overlay nodes to obtain path diversity. Unfortunately, the existing work fall short of least one of the following criteria: (i) application-specific adaptive routing , (ii) scalability, or (iii) fault-tolerance. To the best of our knowledge, communication requirements for the DA application over heterogeneous communication network run by composite public and private operators has not been addressed before.

Contributions

In this paper, we propose an overlay network, *HetGridNet*, that addresses the following requirements of DA applications running on the public heterogeneous network: (1) real-time performance, (2) fault-tolerant communication and (3) E2E QoS-managed delivery. To address these requirements of DA applications, HetGridNet is equipped with the following technical approaches:

(1) The proposed routing mechanism provides real-time communication for DA applications, finding path(s) that meet the requirements of the applications in addition to balancing the traffic on the overlay network. Further, the proposed routing protocol reserves the best paths (in terms of the QoS metrics) for high priority applications by performing priority-based flow allocation. (2) HetGridNet introduces the following fault-tolerance features: multi-path routing over d disjoint paths for only critical applications and periodic heartbeats to detect faulty links in time. (3) HetGridNet's QoS mechanism guarantees E2E QoS-managed delivery across the heterogeneous network and it support diverse and online changeable QoS requirements of DA applications.

Furthermore, its clustering based on AS and two-level overlay network approaches mitigates the overhead of the link state dissemination in the proposed routing protocol and enhances the routing performance. The remainder of the paper is organized as follows. Section II presents the communication

---

Research supported in part by BMBF TUD EC-SPRIDE

Applications	Bandwidth	Latency	Reliability
AMI	10-100 kbps/node, 500 kbps for backhaul	2-15 sec	99-99.99%
Demand Response	14kbps- 100 kbps per node/device	500 ms-sev. min.	99-99.99%
Distribution Energy Resources and Storage	9.6-56 kbps	20 ms-15 sec	99-99.99%
Electric Transportation	9.6-56 kbps, 100 kbps is a good target	2 sec-5 min	99-99.99%
Distribution Grid Management	9.6-100 kbps	100 ms-2 sec	99-99.999%

TABLE I: Smart Distribution Grid Communication Requirements [8]

requirements of the grid followed by Section III detailing the system, traffic and perturbation, models. Section IV, V and VI present HetGridNet's design, routing and QoS approaches. We evaluate HetGridNet in Section VII. Our conclusions and directions for future work appear in Section VIII.

II. COMMUNICATION REQUIREMENTS FOR SMART DISTRIBUTION GRID

This section provides the requirements for the Smart Distribution Grid communication infrastructure, namely: (1) *E2E Latency Guarantee:* Latency is a critical parameter and if the communication delay between devices exceeds the given timeliness threshold, the information has limited utility, and in the worst case, damage might occur in the distribution grid. (2) *E2E Reliability:* According to [8], the reliability of SG is defined as the degree to which a communications system must remain operational. Hence, it is essential that the communication network is reliable for successful and timely message delivery. (3) *Fault-tolerance:* The data of the critical applications has to be accurately delivered, despite network failures. (4) *E2E QoS Guarantee:* Some DA applications, (e.g., including islanding protection) necessitate <1 sec and high priority, while meter-reading acquisition and some other applications are generally anticipated only within a time interval measured in minutes or hours with a lower priority. Moreover, under different conditions some of the applications need different priorities for their traffic according to the usage of data. The U.S. Department of Energy (DoE) has defined communication requirements in smart distribution grid for different functional domain and types of information exchanges as shown in Table 1.

III. MODELS

We now describe the system, data and perturbation models driving our approach.

A. System Model

For DA applications in the distribution grid, we consider a heterogeneous network, which may include both the public Internet and private networks, consisting of many ASes operated by different carriers. Some communication nodes (i.e., sensors and activators) have low computation capacity whereas the other nodes (i.e., substation servers) have high computation capacity and bandwidth with multihoming features. Without loss of generality, we assume that our application scenario has the sensor devices sending the detected data to a local control center and it relays a control data to the corresponding activator(s) if it is necessary.

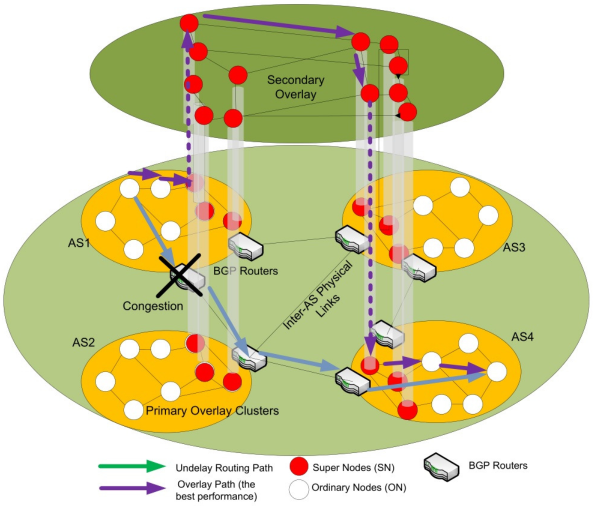

Fig. 1: HetGridNet Architecture

In HetGridNet Architecture, for scalable routing on the overlay network, we cluster the overlay nodes according to their ASes (see, Fig. 1) and each cluster is referred to as the primary overlay cluster. Within each cluster, $(d \geq 2)$ nodes that have adequate bandwidth and computation power are assigned as a Supernode (SN). The other overlay nodes within the clusters are called Ordinary Nodes (ON). A secondary overlay is constructed among these SNs. The SNs act as a gateway between the overlay layers. E2E data transmission on the overlay layers occurs with every node within the primary overlay cluster sending packets to its default ingress SN(s). After the SN receives the packets, they are routed over QoS-satisfied paths on the secondary overlay towards egress SN(s) (to which the destination node belongs) and it delivers the packets to the destination node.

The overlay topology is modeled as a directed graph $G = (V,E)$, where V is the set of overlay node (ON, SN $\in V$) and E is the set of overlay links in the overlay topology. The overlay link between V_i and V_j is denoted as (i, j). While C_i represents current available computation capacity of V_i; B_{ij}, L_{ij}, R_{ij} are respectively available bandwidth, latency and reliability of its adjacent link (i, j). We assume that each node regularly derives this information and relays it to the corresponding overlay node. For instance, a SN relays it to all the other SNs and a ON sends it to its default SN.

We assume that each node in the primary overlay cluster applies to its default SN (assigned by Master SN, cf. Section IV) the R_B, R_L and R_R values respectively (the required node bandwidth, latency and application reliability) in order to select QoS-satisfied path(s) between the source and the destination node. When a message is received by an ingress SN, we employ a Bloom Filter model to find the egress SN to which the destination node belongs to as inspired by [13]. Furthermore, we consider that the bootstrap nodes broadcast the updated bit array of every SNs once the members of the clusters change. The overlay network is expected to have a low churn rate due to the characteristic of the communication nodes.

B. Data Type and Delivery Model

Three types of data exist for DA applications [1]. (D1) Sensing traffic, (D2) Control traffic and (D3) Coordination traffic. All data type can be periodic or aperiodic, and their sizes are typically less then 1 Kb.

Subsequently, we categorize data delivery requirements of the DA applications into three different modes in HetGridNet: (M1) No guaranteed data delivery, (M2) Guaranteed data delivery and (M3) Guaranteed and timely data delivery. In all modes, the packets are formed by employing the UDP protocol. In order to address UDP's fault-tolerance shortcomings and derive the guaranteed data delivery, we propose application-based adaptive ACK schema in M2 and M3. To obtain timely and guaranteed data delivery in M3, multi-routing techniques are employed with disjoint paths.

We employ a priority scale which ranges from very high to low priority (very high, high, medium and low priority). We assume that safety or mission critical applications with low latency and high reliability requirements are assigned to very high priority.

C. Perturbation Model

The communication reliability is affected by both the underlay and overlay network failures. These failures include time-out failures, link failures, and overlay node failures. A time-out failure stems from underlay or overlay network congestion or packet loss etc. A link failure occurs when there is a failure in one of the underlying resource (e.g., the BGP routers, cf. Fig. 1). Finally, the overlay node failure may cause communication disruption if the node is over the defined path between the source and destination node.

IV. HETGRIDNET DESIGN

We now explain the three design stages of HetGridNet i.e., the clustering of nodes and SN selection, the overlay construction, and determining disjoint redundant paths. HetGridNet design aims to (1) mitigate the overhead of the QoS routing which derive real time communication over the public heterogeneous network, and (2) obtain disjoint redundant paths to pave the way for fault-tolerant communication for the time sensitive and high criticality applications.

A. Clustering of nodes and SN selection

To mitigate the overhead of the overlay-based link state routing and to improve routing scalability and performance on the overlay network, HetGridNet clusters nodes depending on their AS and defines a secondary overlay among SNs selected from primary overlay clusters depending on their resources (i.e., computation capacity and outgoing bandwidth). The objective of the secondary overlay employment is to overcome policy based inter-AS routing of BGP that run on its underlying network, and to make routing decisions depending on link cost factors (e.g., latency, throughput) in a scalable manner.

To cluster nodes according to AS and extract AS-AS connection relationships, we utilize the approach from [2]. Accordingly, BGP updates can be regularly accessed by a bootstrap node. An IP prefix to origin AS mapping table is built, and the AS-AS connection relationships extracted from the BGP routing table entries. To construct AS-based clusters, the IP-prefixes are matched with the nodes' IP.

Following the clustering, d supernodes (with one of them as the master supernode) are chosen by the bootstrap depending on their computational capacity and the bandwidth of the outgoing links for each cluster. After this selection process, the SNs receive all SNs' IDs, their members' IDs and the AS-AS connection relationships, whereas ONs acquire only their cluster's SNs IDs.

B. Overlay Construction

In HetGridNet primary overlay clusters, a default SN is selected for each ON by the Master SN depending on its residual bandwidth and the required bandwidth needed for the ON. ONs connect to the default SN over a default link, and to the other $(d-1)$ SNs using $(d-1)$ redundant links. The links between a given ON and the $(d-1)$ SN are referred to as a redundant link taking E2E routing and data delivery into account.

Secondary overlay links are constructed according to the physical links that connect ASes. If an inter-AS physical link exists between two ASes, there are overlay links which connect the SNs from the two ASes.

C. Determining Disjoint Redundant Paths

In order to ensure fault-tolerant communication for the critical applications, HetGridNet aims to define E2E disjoint redundant paths. Path disjointedness is considered as underlay router differentiation of the paths. Firstly path diversity is needed from ON to the SNs over the primary overlay cluster. To this end, each node performs traceroute to every other node in the same cluster to derive underlying routers knowledge between the nodes. The results are relayed by each node to their Master SN. The following process is individually performed for each ON by their Master SN. To differentiate underlay routers of the links between a given ON and d SN, each the redundant link is converted to an indirect path through a relay ON. The selection of the relay ON is performed taking into account underlay router correlation between the default link and the indirect paths. The metric in correlation computation is the number overlapping underlying routers (see [7]). On the other hand, since the secondary overlay is based on physical connectivity of the underlying network, deriving path diversity over secondary overlays does not need additional effort such as obtaining the underlay router knowledge (cf. Section V-B-3).

V. HETGRIDNET ROUTING

In HetGridNet, the overlay layer implements two different routing mechanisms as explained in this section. DA applications mainly contain two data traffic types: periodic and aperiodic data. The following routing mechanisms are for the application that needs the specific periodic data transmission (for the aperiodic message, see Section V-B-3).

A. Routing Towards Supernodes (Primary Overlay Clusters)

In primary overlay clusters, each node directly sends their data to the default SN. However, very high priority messages are relayed over d disjoint paths to its own cluster's d SNs.

B. QoS Routing (Secondary Overlay) (QRSO)

The QRSO protocol strives to find the QoS-satisfied path for each node according to its QoS requirement (e.g., bandwidth, latency, reliability) in addition to traffic balancing over the overlay. Moreover, while selecting paths, it performs priority-based flow allocation to save the *best* path for very high priority applications.

1) Path Selection and Cost Function Definition: In QRSO, we employ the k shortest path (least-cost) routing algorithm (described next section) for path selection between the ingress and the egress SN. We aim to find the least cost (weight) path which meets the QoS requirements in addition to balancing the link load. Hence, we need to define the weights of links and the function which compute the weight of paths for the k shortest path algorithm.

Let the overlay path pass through n SNs (from SN$_s$ to SN$_d$). Proportional Bandwidth Shortest Path (PBSP) [4] defines the path weight function by including the influence of all the concave metrics (e.g., bandwidth, etc.) as: $\sum_i^{n-1}(\frac{B_{i,i+1}}{B_{i,i+1}-R_B} * \frac{C_{i,i+1}}{C_{i,i+1}-R_C})$, where $C_{i,i+1}$ and R_C are residual and required any other concave metric, respectively. The aim of the definition is to maximize the residual bandwidth and other metrics at any link for any path (cf. [4]). However this path weight function does not include the influence of additive metrics (e.g., latency). We sum the additive metrics' influence over the weight of the path as:

$$PathWeight = \partial_{conc}/n * \partial_{add}, \tag{1}$$

where the influence of the concave (∂_{conc}) and the additive (∂_{add}) metrics. We define the influence of the additive metrics (∂_{add}) (latency and reliability) over the weight of the path as;

$$\partial_{add} = (\ell * \Re), \tag{2}$$

where ℓ and $\Re$ are the influence of latency and reliability over the weight of the path respectively.

Firstly, the latency's influence (ℓ) is defined based on the following criteria. Let P_n and P_m be the probability of choosing the paths which pass through n and m intermediate nodes from SN_s to SN_d respectively:

if $\sum_i^{n-1} L_{i,i+1} > \sum_j^{m-1} L_{j,j+1}$ then $P_n < P_m$.

In the definition of the latency's influence (ℓ) , our aim is to minimize the current latency at any link for any path, selecting the minimum latency path, as in PBSP, thus: if $\sum_i^{n-1} L_{i,i+1}< R_L$, $\sum_j^{m-1} L_{j,j+1} < R_L$ and $\frac{R_L}{R_L-\sum_i^{n-1} L_{i,i+1}}$ $> \frac{R_L}{R_L-\sum_j^{m-1} L_{j,j+1}}$ then ($P_n = \frac{R_L-\sum_i^{n-1} L_{i,i+1}}{R_L} < P_m = \frac{R_L-\sum_j^{m-1} L_{j,j+1}}{RL}$). The weight of the paths can be specified as 1 / P . The latency (ℓ) is defined as:

$$\ell = \frac{R_L}{R_L - \sum_i^{n-1} L_{i,i+1}}. \tag{3}$$

Although the reliability is probabilistic metric, it can be converted into additive ones by calculating the logarithm of their product [14]. Based on this concept, similar to the approach for latency, the influence of the reliability over the weight of the path can be defined as:

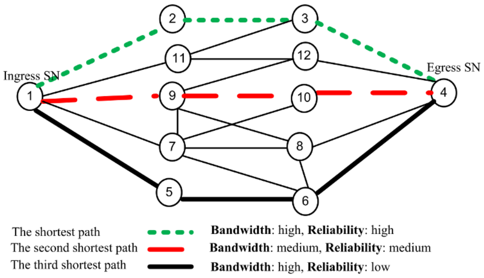

Fig. 2: Priority Based Flow Allocation Example

$$\Re = \frac{logR_R}{logR_R - \sum_i^{n-1} logR_{i,i+1}}. \tag{4}$$

Putting ℓ and $\Re$, defined above, into the equation (2):

$$\partial_{add} = \frac{RL}{RL - \sum_i^{n-1} L_{i,i+1}} * \frac{\sum_i^{n-1} logR_{i,i+1}}{\sum_i^{n-1} logR_{i,i+1} - logR_R}. \tag{5}$$

Finally, ∂_{add} and ∂_{conc} can be put into the equation (1) to obtain the path weight equation as:

$$PathWeight = \sum_i^{n-1}(\frac{B_{i,i+1}}{B_{i,i+1} - R_B})/n *$$

$$* \frac{logR_R}{logR_R - \sum_i^{n-1} logR_{i,i+1}} * \frac{R_L}{R_L - \sum_i^{n-1} L_{i,i+1}}. \tag{6}$$

2) Priority-based Flow Allocation and Data Forwarding: In HetGridNet, after a primary overlay node applies its the QoS requirements to the ingress SN, this SN finds QoS-satisfied path in order to perform flow allocation. However, if we select the shortest path, meeting the requirements, for all application in order to perform flow allocation, the unintended use of the *best* path by low priority applications is unavoidable. To avoid this, the k shortest path, meeting the requirements, are defined between the ingress SN_s and the egress SN_d by using the k shortest path algorithm and our path weight equation (7). Thus QRSO selects from among the k shortest path depending on the application's priority. Figure 2 illustrates a simple case where the ingress SN has two applications. One is protection application with high bandwidth and high reliability requirements and second is video streaming with high bandwidth and low reliability. If the Ingress SN misemploys the shortest path (1-2-3-4) allocating for video streaming, the high priority (supposing reliability is proportional with priority) application may not be satisfied with the selection from the other paths.

For the applications, the corresponding path is selected by the ingress SN between the shortest path and k_{th} path by using the following equation:

$$\lambda = k(1-\rho), z \leftarrow \lfloor\lambda\rceil \tag{7}$$

where (ρ) is the priority which fall in the range of [0-1], e.g., 0 is low priority and 1 is very high priority. Using the equation, z_{th} path is defined by the ingress SN for the flow allocation.

After the path selection, a flow allocation message is sent to the egress SN. This message contains route information and a Flow ID. All SNs that are over the path record this Flow ID and its next hop to a forwarding table. Thus, all packets that belongs to this flow are forwarded by intermediate nodes (barring significant link changes). This flow allocation is periodically updated depending on updated link state information.

3) Fault-tolerance: QRSO employs multi-path routing over disjoint path for very high priority application on secondary overlay. Hence, in each cluster, the Master SN selects disjoint paths for each SN after matching its cluster SN with the destination SN, by rendering the highest bandwidth and the disjointedness trade-off. Thus, when an ingress SNs receive high priority packets, they send the packets over the disjoint paths defined by the Master SN toward the egress SNs which deliver the packets the destination node. In this routing mechanism, we prefer source routing rather than flow allocation using flow ID. Moreover, we use this mechanism in both, low and high priority applications, that require aperiodic data delivery.

C. Failure Recovery

For outage detection, each node periodically uses an active probing mechanism. Over the secondary overlays if a failure or heavy congestion is detected by any SN, the SN broadcasts this information to all SNs in order to redefine their own paths that pass through the failed link. Meanwhile, the SN strives to find a path to bypass the failed link for the ongoing flows that use the link. On the other hand, if primary overlay node detects a failure on the link, the nodes switches the default link with one of redundant path for constant data delivery.

D. The Acknowledgment mechanism

For the messages that require delivery guarantee, HetGridNet supports a adaptable acknowledgment mechanism by allowing the applications to configure retransmission time and number depending on message priority and time-sensitiveness.

E. HetGridNet QoS

In HetGridNet QoS mechanism, we suppose that the priorities of the applications are delivered online to the supernodes by corresponding control center at each change. According to the priorities of the applications, supernodes prioritize the application traffic and performs online adaptable QoS management of the data traffic across the public heterogeneous network.

VI. HETGRIDNET EVALUATION

HetGridNet is implemented using the OverSim [16] and INET framework which runs on the OMNeT++ [17]. In this section, we first introduce underlay topology and background traffic model, followed by overlay and performance parameters and metrics. Finally, we present our simulation experiments. The goal of our simulation-based study is to evaluate HetGridNet regarding the following aspects compared to the Shortest Path Algorithm (SPA) in the overlay network. (1) QoS-satisfaction of the application on the ONs, and (2) Fault-tolerance.

A. Underlay Network Topology

App.	**Msg. size**	**Param.**	**Priority(p)**	R_R	R_L	**Delivery Mode**
App1	32 B	1 event /15s	Medium (0.5)	Medium(99%)	High (<2s)	M2
App2	32 B	1 event /20s	Very high (1)	High(99.90%)	Low (<150)	M3

TABLE II: PERFORMANCE EVALUATION PAPAMETERS

To build an internet-like underlay topology, we generate random hierarchical topologies using BRITE [15]. The topology consists of 10 nodes (routers) on the AS level and 10 nodes (edge routers) under the each AS node with an edge density varying from 2 to 5. For inter-AS and intra-AS networks, two bandwidth settings are employed: all links are either OC3 (i.e., 155 Mbps) or OC48 (i.e., 2.48 Gbps). The propagation delay of each link is a random number between 0-10 msecs subject to a uniform distribution.

B. Background Traffic

To evaluate HetGridNet's behavior in the case of dynamic latency and bandwidth in the underlay network, we generate background traffic random load across the network during simulation. To produce this background traffic, we employ 100 servers (each one connects to each edge router) that send 1-100kb sized packets per sec to a random server in every seconds.

C. Overlay Network and Traffic Demands

In the simulation, $|V|$=1000, (|SN|=20 and |ON|=980), and they are evenly distributed to 10 ASes. The AS-based clusters have two (d=2) SNs and their computation capacities are sufficient and larger than ONs. The outgoing of SNs and ONs is 12Mbps (ADSL) and 171kbps (GPRS), respectively.

According to DA applications' data traffic requirements [1], we use the two applications models from Table II. Each overlay node randomly employ with one of the application. They also randomly select a destination node to send the application's messages.

D. Metrics

We evaluate HetGridNet based on QoS-satisfaction of the application on each node and the fault-tolerance. QoS-satisfaction of a given node is calculated based on its data delivery statistics: latency and loss rate. A dropped or timed out message is referred to as unsuccessful message delivery. The deliveries satisfy the application's requirements in case of successful messages divided by sent messages is above in the application reliability rate. Moreover, to assess the fault-tolerance of the protocols, bit error rate (BER) setting of links is switched from the range of (1e-10, 1e-7) to (1e-10, 1e-3).

E. Simulation Results and Discussion

Figure 3 depicts the QoS-satisfaction rate of the applications running on the overlay nodes by comparing QRSO and SPA algorithms in two different application requirements. In the figure, the lines in the middle of the boxes present the mean value of the ten repeated experiments while the boxes show the 95% confidence intervals of the experiments. In addition, the whiskers depict minimum and maximum values of the experiment results. To quantify the QoS-satisfaction factor on a given application running on a overlay node, QoS-satisfaction rate (QSR) is defined as

$$QSR = \frac{SentMessages - DroppedOrTimedoutMessages}{SentMessages} \tag{8}$$

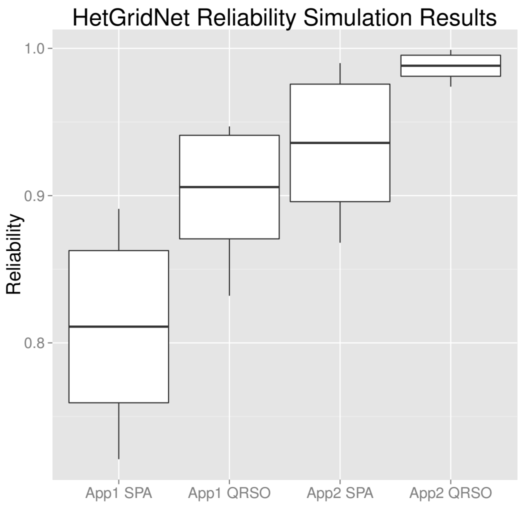

Fig. 3: QoS-satisfied Rate comparison of the algorithms in the the range of (1e-10, 1e-7) BER

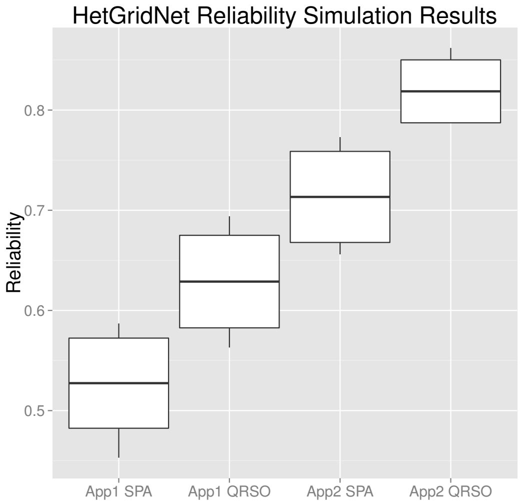

Fig. 4: QoS-satisfied Rate comparison of the algorithms in the the range of (1e-10, 1e-3) BER

Since the Internet infrastructure contains physical links which have diverse reliability rate, in many situations, SPA cannot meet the application requirements. However, we can observe that QRSO greatly improves the QSR, as QRSO employs the weight function (6) including the reliability constraint. In addition, bandwidth and latency constraints in the function contribute decrease of the time out message number. Although App2's reliability and latency requirements are so strict, QRSO significantly provides a better QSR for App2, thanks to our priority based flow allocation mechanism.

Figure 4 shows the QSR of the algorithms after worsening the BER from the range of (1e-10, 1e-7) to (1e-10, 1e-3). We can observe that the reliability influence on the weight function obtains better QSR even in the present of high BER.

VII. CONCLUSION

HetGridNet provides reliable and real-time communication on heterogeneous public and private network, considering the DA applications' requirements. It selects the overlay nodes with the most adequate resource provisioning to manage inter-AS communication rather than place dedicated server into each domain and needs local underlay knowledge to enable reliable communication across the network. Moreover, QRSO (employing on the secondary overlay) find the *best* path considering bandwidth, latency, and reliability requirements of the applications in proportional manner. It uses priority-based flow allocation to reserve the *best* path for high critical applications in addition to the overlay based QoS mechanism. The simulation result demonstrates that HetGridNet provides about 15%- 20% benefit of QoS-satisfied rate using the QRSO algorithm compared to the SPA algorithm.

These result suggest that HetGridNet is a suitable architecture for smart distribution applications working on heterogeneous public or private networks. In addition, we plan to implement and evaluate HetGridNet in the PlanetLab testbed environment.

REFERENCES

[1] Y. Qiang, et al., "Communication infrastructures for distributed control of power distribution networks," IEEE Trans. Ind. Informatics, vol. 7, no. 2, pp. 316-327, 2011.

[2] S. Ren, et al., "Asap: an as-aware peer-relay protocol for high quality voip," In Proc. of Int. Conf. on Distributed Computing Systems, pp. 70-79, 2006.

[3] P. M. Kanabar, et al., "Evaluation of Communication Technologies for IEC 61850 Based Distribution Automation System with Distributed Energy Resources," Proc. of the IEEE PES General Meeting, Calgary, pp. 26- 30, 2009.

[4] Z. Li and P. Mohapatra, "QRON: QoS-aware routing in overlay networks," IEEE Journal on Selected Areas in Communications, vol. 22, no. 1, pp. 29-40, 2004.

[5] M. P. Comech, et al., "Protection in distributed generation."Distributed generation, chap. protection in distributed generation In-Teh, Vukovar, Croatia, pp. 289-310, 2010.

[6] K. C. Budka, et al., "Communication network architecture and design principles for smart grids" Bell Labs Tech. Journal., vol. 15, no. 2, pp. 205-228, 2010.

[7] J. Han, et al., "Enhancing end-toend availability and performance via topology-aware overlay networks," Computer Networks, vol. 52, Issue 16, pp. 3029-3046, 2008.

[8] US Dept. of Energy, "Communications Requirements of Smart Grid Technologies," tech. rep., 2010.

[9] A. Vallejo, et al., "Next-generation QoS control architectures for distribution smart grid communication networks," IEEE Commun. Mag., vol. 50, no. 5, pp. 128-134, 2012.

[10] D. E. Bakken, et al., "Smart generation and transmission with coherent, real-time data," in Proc. of the IEEE vol. 99, no. 6, pp. 928-951, 2011.

[11] D. Andersen, et al., "Resilient overlay networks," vol. 35, no. 5, pp. 131-145, 2001.

[12] G. Wang, et al., "An efficient relay node selection scheme to improve the performance of P2P-based VoIP applications in Chinese Internet," Multimedia tools and applications vol. 64, no. 3, pp. 599-625, 2013.

[13] B. Y. Zhao, et al., "Brocade: Landmark Routing on Overlay Networks, in Proc. of IPTPS," pp, 33-34, 2002.

[14] D. Ilie, and A. Popescu, "Unicast QoS routing in overlay networks," Network Performance Engineering, pp. 1017-1038, 2011.

[15] A. Medina, et al., "BRITE: An approach to universal topology generation," in Proc. of MASCOTS, pp. 346, 2001.

[16] I. Baumgart et al., "OverSim: A Flexible Overlay Network Simulation Framework," in Proc. GI at INFOCOM, pp. 79-84, 2007,

[17] G. Pongor, "OMNeT: Objective Modular Network Testbed," in Proc. of MASCOTS, pp. 323–326, 1993