Analyzing Storage Requirements of the Resilient Information-Centric SeDAX Architecture

Michael Hoefling, Cynthia G. Mills, and Michael Menth
University of Tuebingen, Chair of Communication Networks, Tuebingen, Germany
Email: {hoefling,menth}@uni-tuebingen.de, cynthia.mills@student.uni-tuebingen.de

***Abstract*—We develop and analyze algorithms that reduce the storage capacity required by SeDAX (Secure Data-centric Application eXtensible) in the presence of simultaneous node failures. The SeDAX infrastructure for smart grids uses data redundancy for a high level of reliability. It is an information-centric approach using resilient data forwarding in a Delaunay triangulated overlay. While SeDAX's data forwarding scheme is well understood, there is no study that considers the SeDAX storage capacity necessary to survive multiple node failures. Our results are compared with the theoretical lower bound of SeDAX and the lower bound of an idealized storage system. The presented algorithms can be used to reduce storage requirements of SeDAX in practice.**

I. INTRODUCTION

Today, smart grid refers to the next-generation power grid designed to enhance its resilience to power flow disruptions, improve energy efficiency, and reduce carbon emissions. A main obstacle to the deployment of smart grid applications is the limited scalability, reliability, and security of todays utility communication infrastructure. The NIST (National Institute for Standards and Technology) working group on smart grid [3] has identified reliability requirements for smart grid communication flows.

The recently published SeDAX architecture [1] aims to provide such a reliable platform. SeDAX applies the emerging information-centric networking (ICN) paradigm to the electric utility network of sensors and controls for electricity generators, consumers, and brokers. A Delaunay triangulated (DT) overlay provides resilient name-based geographic data forwarding. SeDAX's data forwarding has been well analyzed by the authors [1]. Furthermore, data redundancy provides resiliency when node failures occur. However, a study that considers the SeDAX system storage necessary to survive multiple node failures without storage shortages is still needed. This paper addresses that gap.

We review relevant aspects of the SeDAX architecture in Section II and discuss related work in the context of ICN and data placement in Section III. Section IV derives formulas for SeDAX system storage and suggests a Monte-Carlo based optimization for SeDAX node placement to minimize storage requirements. Section V presents theoretical lower bounds for SeDAX's storage requirements and for an idealized storage system, and compares them with the simulative results, leading to our conclusions in Section VI.

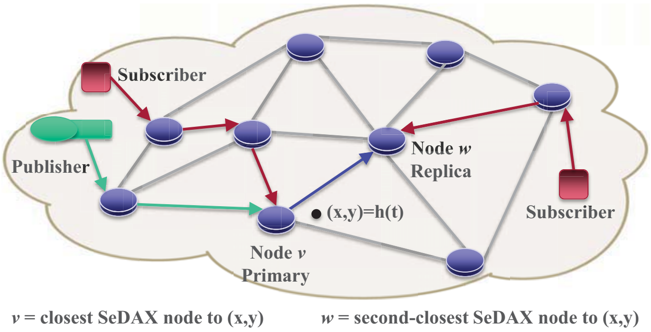


Fig. 1. Topic-group communication in SeDAX uses geographic forwarding.

II. THE SEDAX ARCHITECTURE

SeDAX organizes information into *topics* that are stored in a network overlay based on a DT graph with SeDAX *nodes* as vertices and transport connections between neighboring nodes as edges, as shown in Figure 1. We denote T as the set of all topics and V as the set of all SeDAX nodes.

Information contributors, such as sensors, *publish* data to topics. Consumers *subscribe* to topics so that new topic data is automatically forwarded to them. Since topic data may expire, SeDAX nodes require only sufficient capacity to store current, i.e., non-expired, topic data and are not intended for archival purposes. Therefore, limited storage is sufficient for the data of a topic $t \in T$ and is given by $c(t)$ storage units. Operational topic volumes have not yet been determined.

A characteristic feature of SeDAX is its mapping of *topics* to *nodes*. A geographic hash function (GHF) $h(t)$ calculates Euclidean *coordinates* in the plane for each topic $t \in T$. Each SeDAX node $v \in V$ is also associated with a planar coordinate that reflects its position in the overlay network. The Euclidean metric $d(v,t)$ determines the distance between topics and nodes based on their coordinates.

The primary copy of a topic's data and subscriptions is stored on the SeDAX node closest to a topic's coordinates and a backup is stored on the second-closest SeDAX node. Figure 1 illustrates the geographic forwarding approach used in SeDAX. The DT overlay structure guarantees that messages destined for the topic coordinate $(x,y) = h(t)$ are iteratively relayed to the active (non-failed) SeDAX node v whose coordinates are closest to the destination. If the closest or second-closest node fails, the other replicates the topic's subscriptions

IEEE, 4^{th} International Conference on Smart Grid Communications (SmartGridComm), Vancouver, Canada, October 2013

and data to the originally third-closest (now second-closest) node.

Node failures are detected by broken or timed-out TCP connections between SeDAX nodes. After a node failure, the overlay reconfigures itself to restore the overlay DT properties and heal the forwarding. The resilient forwarding and the concept of primary and backup nodes constitute a simple resilience concept in SeDAX.

SeDAX is highly scalable because publishers and topic stores need not maintain TCP connections with each other. Each SeDAX node holds TCP connections only to its immediately neighboring nodes and publishers.

III. RELATED WORK

SeDAX [1] builds upon prior work in the area of publish-subscribe [4] and ICN [5]. Most existing architectures are based on distributed hash tables (DHTs) like Chord [6] or CAN [7] for topic resolution and use an overlay for data forwarding to the topic stores. PSIRP/PURSUIT [8], 4WARD/SAIL [9], and NDN/CCNx [10], [11] are ICN architectures which use DHTs and the publish-subscribe paradigm. Alternative approaches mostly differ in the way topic names are resolved and data is forwarded. DONA's [12] topic resolution system consists of a hierarchically organized network of resolution handlers. LIPSIN [13] uses bloom filters to resolve topic names and find the topic stores.

In contrast, the SeDAX architecture specifically addresses the requirements of the smart grid. SeDAX's unique security framework [2] covers both information and data security considerations for SeDAX as a cyber-physical system. SeDAX adopts the recommendation of [14] to use topic names as input for the GHF instead of publisher names for sensor networks. SeDAX requires no mapping system for topic resolution because static GHF coordinates allow publishers and subscribers to simply and autonomously calculate a topic's coordinate. SeDAX adds the DT scheme to the overlay which enables automatic resilience. The use of a single neighboring backup and a self-healing network means that in case a failure, a neighbor can immediately respond to a request rather than forwarding it to a distant backup whose state is unknown. This does not preclude the use of more complex load-balancing schemes for archival data. Other load balancing schemes for wireless sensor networks organize the distribution of data into a hierarchy [15] or modify the GHF to include attributes such as temporal information [16]. QoS constraints for replication in these more complex topologies with hierarchical data stores are discussed in [17], [18].

IV. SIMULATIVE STORAGE ANALYSIS

This section presents a simulative analysis of storage requirements for SeDAX nodes. First, the performance metrics of interest are introduced. Then, the experiment setup for the evaluation is given together with an optimization scheme for SeDAX node placement. The simulation results show storage requirements for SeDAX with optimized SeDAX node placement.

A. Performance Metrics

We denote S as the set of all considered failure scenarios. A failure scenario $s \in S$ represents a set of failed nodes including the failure-free case. Given a maximum number n_{fail}^{max} of failed nodes, S contains all combinations of up to n_{fail}^{max} simultaneously SeDAX node failures. Next, we define performance metrics that can be applied when the coordinates of all topics $h(t)$ and nodes as well as each topic's storage requirements $c(t)$ are known.

We define several loads per SeDAX node for failure scenario s. The *primary load* $L_{prim}(v,s)$ gives the sum of the storage requirements of topics for which v is the closest node under failure scenario s.

$$T_{prim}(v,s) = \{t \in T : \forall w \in V \setminus \{v,s\}, d(v,t) < d(w,t)\}$$

$$L_{prim}(v,s) = \begin{cases} 0, & \text{if } v \in s \\ \sum_{t \in T_{prim}(v,s)} c(t), & \text{else.} \end{cases} \tag{1}$$

The *secondary load* $L_{sec}(v,s)$ is the sum of the storage requirements of topics for which v is the second-closest node under failure scenario s.

$$\begin{aligned} T_{sec}(v,s) = \{t \in T : \exists^{=1} u \in V \setminus \{v,s\} \forall w \in V \setminus \{v,u,s\}, \\ (d(u,t) < d(v,t)) \land (d(v,t) < d(w,t))\} \end{aligned}$$

$$L_{sec}(v,s) = \begin{cases} 0, & \text{if } v \in s \\ \sum_{t \in T_{sec}(v,s)} c(t), & \text{else.} \end{cases} \tag{2}$$

The *node load* $L(v,s)$ is the sum of the storage requirements of topics for which v is the primary or secondary node under failure scenario s.

$$L(v,s) = L_{prim}(v,s) + L_{sec}(v,s) \tag{3}$$

Based on the node loads, we define capacity requirements for SeDAX nodes and the system. The *node capacity requirement* $c_{node}(v)$ specifies the minimum capacity required to store the node load in all failure scenarios $s \in S$.

$$c_{node}(v) = max_{s \in S}\left(L_{all}(v,s)\right) \tag{4}$$

The *maximum node capacity requirement* c_{node}^{max} is defined as the largest capacity requirement $c_{node}(v)$ of any node $v \in V$ in all failure scenarios $s \in S$.

$$c_{node}^{max} = max_{v \in V}\left(c_{node}(v)\right) \tag{5}$$

It specifies the minimum storage requirements of nodes if all nodes are provisioned uniformly or *homogeneously*, i.e., all nodes have equal storage.

The *system capacity* c_{sys} specifies the SeDAX network-wide storage required to survive all failures $s \in S$ without storage shortages if each node is individually or *heterogeneously* provisioned with its minimum needed storage.

$$c_{sys} = \sum_{v \in V} c_{node}(v) \tag{6}$$

Capacity is given in storage units. To generalize results, we express them relative to the system load c_{load}, i.e., the sum of the storage requirement for all topics $c_{load} = \sum_{t \in T} c(t)$.

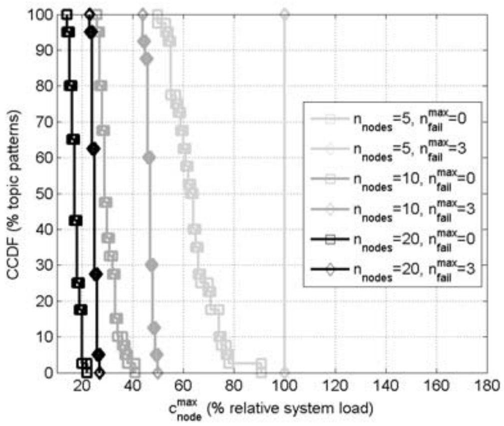


Fig. 2. Impact of the number of nodes n_{nodes} and number of failed nodes n_{fail}^{max} on the maximum node capacity c_{node}^{max} under optimized node placement.

As an example, the required system capacity is 200% relative to the system load when each topic is stored on exactly two nodes in the failure-free scenario, independent of node and topic coordinates.

B. Experiment Setup and Optimization of Node Placement

To evaluate storage requirements for SeDAX under failures, we choose a square plane as coordinate space and create topic and node patterns. We generate random coordinates for $n_{topics} = 100$ topics and for $n_{nodes} = \{5, 10, 20\}$ nodes. Then we calculate the performance metrics for up to $n_{fail}^{max} = 3$ simultaneous node failures.

We perform Monte Carlo optimization of node placement to reduce storage requirements. We produce $n_{patterns}^{node} = 200$ different node patterns and choose the one that requires least system capacity. The results of these experiments depend on the topic pattern. Therefore, we repeat them for $n_{patterns}^{topic} = 40$ different topic patterns and express the results as complementary cumulative distribution functions (CCDFs) based on the topic patterns. To simplify the analysis, we set $c(t)$ to one storage unit, but this is not a constraint for the presented optimization and evaluation framework.

C. Storage for Optimized Node Placement

1) Maximum Node Capacity Requirements: We first assume that all nodes in SeDAX are provisioned with the same amount of storage. In order to survive node failures without storage shortage, all nodes need at least the maximum node capacity requirements c_{node}^{max} as defined in Equation 5. Therefore, we optimize the node placement to minimize c_{node}^{max}.

Figure 2 shows the CCDF of the results of the maximum node capacity requirements for the optimized node placement. We interpret the figure as follows: for each maximum node capacity requirement x on the x-axis, the y-axis gives the percentage of topic patterns whose maximum node capacity requirements X is greater than x.

We observe that the maximum node capacity requirements decrease with an increasing number of SeDAX nodes in the system. This is a trivial result: as the number of topics and their data volume is the same in all experiments, the average load per node is inversely proportional to the number of nodes n_{nodes}, at least for $n_{fail}^{max} = 0$.

We recognize that the maximum node capacity depends on the specific topic pattern. For $n_{nodes} = 20$ nodes, the maximum node capacities range in the failure-free case between 14% and 22%. If up to three nodes fail, the maximum node capacities range between 23% and 27% relative system load. More storage capacity is needed for the SeDAX system to survive additional node failures without storage shortages.

When all nodes in a SeDAX system are provisioned homogeneously, the system-wide capacity requirement is $n_{nodes} \cdot c_{node}^{max}$. For $n_{nodes} = 20$ nodes, a system-wide capacity between 460% and 540% is required.

2) System Capacity: Node-specific storage provisioning is an alternative to homogeneous provisioning. That means, each node is provisioned with its individual node capacity $c_{node}(v)$ to survive up to a given number of node failures n_{fail}^{max} without storage shortages. We now optimize the placement of SeDAX nodes to minimize c_{sys}.

Figure 3 shows the CCDF of the system capacities for the optimized node placement whose mean values are summarized in Table I.

Figure 3 illustrates that the SeDAX system requires significantly more storage to survive up to n_{fail}^{max} node failures compared to the failure-free case. We observe that the system capacity depends on the topics patterns. This is because the backup capacity can be shared more efficiently for some topic patterns than for others. For 20 nodes, the required system capacity is between 254% and 263% for $n_{fail}^{max} = 1$, between 311% and 327% for $n_{fail}^{max} = 2$, and between 368% and 383% for $n_{fail}^{max} = 3$.

The figure further shows that the required system capacity is about the same for $n_{nodes} = 10$ and $n_{nodes} = 20$ nodes and only for a very small number of nodes like $n_{nodes} = 5$, the relative system capacity is clearly larger.

Node-specific storage provisioning leads to significant storage savings compared to homogeneous node storage provisioning. For 20 nodes and a maximum number of $n_{fail}^{max} = 3$ failed nodes, savings up to

$$\frac{n_{nodes} \cdot c_{node}^{max} - c_{sys}}{n_{nodes} \cdot c_{node}^{max}} = \frac{540\% - 368\%}{540\%} \approx 32\%$$

are possible. In other words, homogeneous node storage provisioning requires 68% more storage than node-specific storage provisioning to provide the same level of storage shortage protection.

The outcome of the optimization may seem difficult to implement as node placement in practice is typically determined by operational necessities. However, the assignment of virtual coordinates and their use for the DT overlay combines arbitrary physical placement of nodes with the use of optimized coordinates of SeDAX nodes. The drawback of that approach may be longer path in the DT overlay.

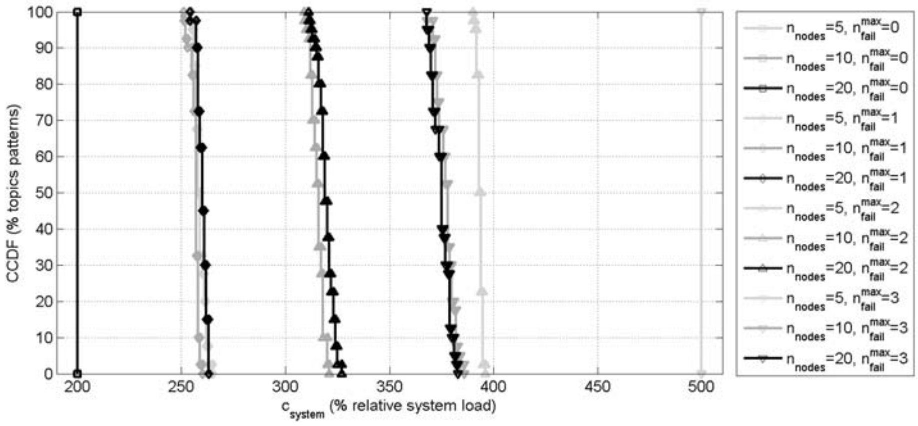
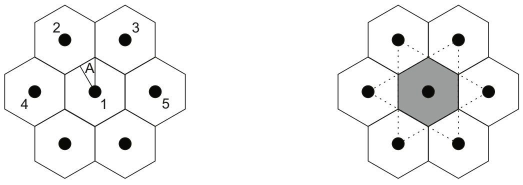


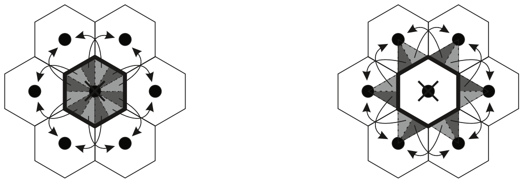
Fig. 3. Impact of the number of nodes n_{nodes} and number of failing nodes n_{fail}^{max} on the system capacity c_{sys} under optimized node placement.



(a) The five closest SeDAX nodes of topic area A.

(b) The node in the middle is closest to the gray area and second-closest to the dashed triangles.

Fig. 4. Supporting figures for the analysis.



(a) Topic areas reassigned to previously second-closest nodes.

(b) Topic areas reassigned to previously third-closest nodes.

Fig. 5. Reassignment of topic area responsibilities if one SeDAX node fails.

V. ANALYTICAL LOWER BOUNDS

We derive lower bounds for SeDAX system capacity requirements that could suffice under optimal conditions. Then, we calculate lower bounds for an idealized storage system. Numerical results are compared with those from simulations.

A. Bounds for SeDAX

In a SeDAX system, the overall storage requirements are smallest when the same amount of data is distributed equally across all SeDAX nodes in the failure-free case and each node shares its load equally among the maximum number of equidistant neighbors. This consideration is the basis for the following analysis.

We consider an infinite plane. A GHF maps a vast number of topics with equal storage requirements evenly over this plane. Since a triangular node arrangement maximizes the number of equidistant neighbors, we use it for the placement of SeDAX nodes.

The *topic area* for which a SeDAX node is the closest node thus forms a hexagon with six adjacent neighbors, as shown by the gray area in Figure 4(a). For area A these nodes are listed in order of proximity. Normally, a topic that maps to area A is assigned to the closest node (node 1) as primary, and its secondary to the second-closest node (node 2). When one of these nodes (node 1 or 2) fails, the third-closest node (node 3) becomes the secondary node. When two of the three closest nodes fail, the affected topic is stored on the fourth-closest node (node 4). For larger numbers of adjacent node failures, the topic responsibility is shifted in the same way.

1) Failure-Free Condition: For all topics that the GHF hashes into a hexagon, see the gray area in Figure 4(b), the primary node is located in the center. We denote the load created by the topics located in a single hexagon as 100% load. Due to the assumption that topics are evenly distributed over the plane, each node carries 100% primary load. There are six triangular areas adjacent to this hexagon. They form the area for which the central node is second-closest (see the areas bounded by the dashed lines in Figure 4(b)) and thereby contribute another 100% load to the central node. Thus, each node carries 200% load in the failure-free scenario.

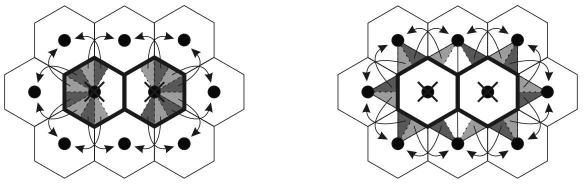
2) Single Node Failures: In Figure 5(a), the center node serves as primary node for the topics mapped to the shaded triangles. When it fails, the secondary nodes take over as primary and the topics are reassigned to new secondary nodes as indicated by the arrows. The load of one triangle corresponds to a load of $\frac{1}{12} \cdot 100\%$.

In Figure 5(b), the center node serves as secondary node for the topics mapped to the shaded triangles. When it fails, new secondary nodes are reassigned to the topics that are mapped to the respective triangles. These nodes are indicated by the arrows.

In both figures together, we count four arrows towards the failed node's neighbors. Thus, each of those nodes receives an additional load of $\frac{4}{12} \cdot 100\% \approx 33.3\%$ so that it must carry an overall load of 233.3%.

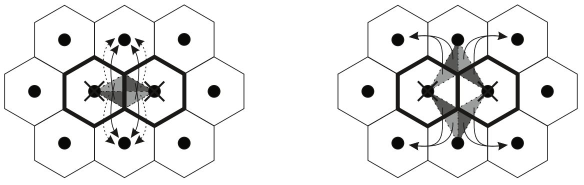
TABLE I
SYSTEM CAPACITY REQUIREMENTS FOR UP TO n_{fail}^{max} NODE FAILURES: SIMULATION RESULTS AND ANALYTICAL LOWER BOUNDS FOR SEDAX TOGETHER WITH LOWER BOUNDS OF AN IDEALIZED STORAGE SYSTEM.

c_{system}	$n_{nodes} = 5$		$n_{nodes} = 10$		$n_{nodes} = 20$		Lower bounds for SeDAX
	SeDAX simulation	**Idealized system**	**SeDAX simulation**	**Idealized system**	**SeDAX simulation**	**Idealized system**	
$n_{fail} = 0$	200%	200%	200%	200%	200%	200%	200%
$n_{fail} = 1$	$255\% - 265\%$	250%	$251\% - 260\%$	222%	$254\% - 263\%$	211%	233.3%
$n_{fail} = 2$	$390\% - 396\%$	333%	$309\% - 321\%$	250%	$311\% - 327\%$	222%	266.7%
$n_{fail} = 3$	500%	500%	$368\% - 386\%$	285%	$368\% - 383\%$	236%	316.7%



(a) When only the closest node to an area fails, its load is addded to the third-closest nodes.

(b) When only the second-closest node to an area fails, its load is added to the third-closest nodes.



(c) When the first- and second-closest nodes to an area fail, its load is added to the third- and fourth-closest nodes.

(d) When the first- and third-closest nodes or the second- and third-closest nodes to an area fail, its load is added to the fourth-closest node.

Fig. 6. Reassignment of topic area responsibilities if two adjacent SeDAX nodes fail.

3) Double Node Failures: We analyze the cases in which the failed nodes are adjacent to each other, separated by exactly one node, or separated by more than one node.

Two adjacent nodes fail. We use the same approach as for the single node failure in Section V-A2 to analyze the failure of two adjacent nodes. Figure 6(a) shows the areas that lose their closest node, but not their second- and third-closest nodes. Thus, a copy of the topics mapped to these areas is added to the third-closest nodes as indicated by the arrows in the figure. Figure 6(b) shows the areas that lose their second-closest nodes, but not their closest and third-closest nodes. Thus, a copy of the topics mapped to these areas is also added to the third-closest nodes as indicated in the figure. Figure 6(c) shows the areas that lose their closest and third-closest node or their second- and third-closest node, but not their fourth-closest node. Thus, a copy of the topics mapped to these areas is added to the fourth-closest nodes as indicated in the figure. Figure 6(d) shows the areas that lose their closest and second-closest node, but not their third- and fourth-closest nodes. Thus, a copy of the topics mapped to these areas is added to both the third- and fourth-closest nodes.

Adding up all reassignments of topic area responsibilities, we see that neighboring nodes of the failed nodes receive additional load from 4, 6, or 8 triangles, which results in a maximum additional load of $\frac{8}{12} \cdot 100\% \approx 66.7\%$. The most heavily loaded nodes are the direct neighbors of the two failed nodes; they must be able to carry a load of up to 266.7%.

Two nodes fail with one node in between. If two non-adjacent nodes fail that are separated only by a single intermediate node, this node receives 33.3% additional load from each of its failed neighbors. This is the worst case which amounts to a maximum load of 266.7%.

Two nodes fail with more than one node in between. If two non-adjacent nodes fail that are separated by more than a single intermediate node, their neighboring nodes receive additional load only from one of the failed nodes. Therefore, the maximum load is 33.3% like in the single node failure scenario.

4) Triple Node Failures: For the sake of brevity, we consider only the worst case in terms of additional load. When three contiguous neighbors of a node fail, the node next to the three failed nodes needs to carry at most a load of 316.7%.

5) Discussion: The analytical lower bounds for system capacity requirements in SeDAX are significantly lower than the simulation results of Section IV which are summarized in Table I. The difference is caused by the fact that the topics in our simulation are less fine-grained and not evenly distributed over the plane, which is more realistic. Moreover, our optimization method could be further tuned.

B. Bounds for an Idealized System

In an idealized storage system, each topic's data is simultaneously stored on both a primary and a secondary node. When one of these nodes fails, its topic data is instantaneously replicated to yet another node so that two copies of the same topic are always available in the system. When a node or topic is added or removed, its topic data is distributed evenly over all nodes. This idealized load distribution leads to theoretical minimum storage requirements.

1) Analysis: Due to the idealized load distribution, each of the n_{nodes} storage nodes carries $\frac{200\%}{n_{nodes}}$ of the system load. When n_{fail}^{max} nodes fail, each of the remaining $n_{nodes} - n_{fail}^{max}$ nodes now carries $\frac{200\%}{n_{nodes} - n_{fail}^{max}}$ of the system load so that the network-wide system capacity requirement of the idealized storage system c_{sys}^{ideal} is defined as

$$c_{sys}^{ideal} = \frac{n_{nodes}}{n_{nodes} - n_{fail}^{max}} \cdot 200\%. \qquad (7)$$

2) Discussion: We compare the system capacity requirements of the idealized storage system with simulation results and the lower bounds for SeDAX in Table I. While the idealized storage system uses only 36% extra system capacity to provide enough capacity to accommodate backup copies if up to $n_{fail}^{max} = 3$ nodes fail, SeDAX requires $168\% - 186\%$ extra capacity. In contrast to SeDAX, the idealized storage system leverages perfect load balancing, so its capacity requirements are independent of topic coordinates and node placement, but its performance cannot be achieved in practice.

The lower bounds for SeDAX are derived for an infinite plane with an infinite number of nodes. We compare them with the system capacity requirements of the idealized storage system. For $n_{fail}^{max} = 3$, we have 116.7% extra capacity compared to 36% extra capacity for $n_{nodes} = 20$ nodes. Even though the lower bounds for SeDAX were calculated for optimal conditions, it is still considerably less efficient than the idealized storage system. We see this deviation because SeDAX cannot efficiently distribute capacity. When a node fails, only its closest neighbors copy its data and provide backups; available capacity on distant nodes cannot be used for that purpose.

VI. CONCLUSIONS

In this paper, we presented formulas for resilient provisioning of SeDAX so that up to n_{fail}^{max} node failures can be survived without shortage of storage. We further developed a simple Monte-Carlo optimization for node placement in SeDAX to minimize storage requirements. The node placement optimization can also take uneven topic distributions into account by using the actual storage requirements of each topic. SeDAX can use the optimized virtual coordinates to configure the network.

We evaluated the capacity requirements of SeDAX with optimized node placement for homogeneous and heterogeneous node provisioning. The latter requires significantly less storage. In general, storage requirements depend on topic patterns. We derived the least storage requirements of SeDAX under optimal conditions and showed that they far exceed those of an idealized storage system. The reason is the inflexibility of the topic location in SeDAX.

Although efficient use of available storage is not the primary goal of SeDAX, our optimization improves the resource management of SeDAX while maintaining its compelling properties, namely scalability, automatic resilience, and security. An alternative to explore would be to keep the coordinates of the nodes and optimize the placement of topics, which would require larger architectural modifications to SeDAX.

ACKNOWLEDGMENT

The research leading to these results has received funding from the European Community's Seventh Framework Programme FP7-ICT-2011-8 under grant agreement n° 318708 (C-DAX). The authors alone are responsible for the content of this paper.

The authors thank Young-Jin Kim, Marina Thottan, Peter Hauck, Marcel Mampaey, Alfons Martin, Florian Heimgaertner, and Mark Schmidt for valuable input and stimulating discussions.

REFERENCES

[1] Y.-J. Kim, J. Lee, G. Atkinson, H. Kim, and M. Thottan, "SeDAX: A Scalable, Resilient, and Secure Platform for Smart Grid Communications," *IEEE JSAC*, vol. 30, no. 6, 2012.

[2] Y.-J. Kim, V. Kolesnikov, H. Kim, and M. Thottan, "Resilient End-to-End Message Protection for Large-scale Cyber-Physical System Communications," in *IEEE SmartGridComm*, Nov. 2012.

[3] "D1 Requirements for different Smart Grid Applications, SG Network Systems Requirements Specification V4.0," NIST Smart Grid Interoperability Panel PAP01: Role of IP in the Smart Grid, Nov. 2009.

[4] P. T. Eugster, P. A. Felber, R. Guerraoui, and A.-M. Kermarrec, "The Many Faces of Publish/Subscribe," *ACM Computing Surveys*, vol. 35, no. 2, 2003.

[5] A. Ghodsi, T. Koponen, B. Raghavan, S. Shenker, A. Singla, and J. Wilcox, "Information-Centric Networking: Seeing the Forest for the Trees," in *ACM HotNets*, Nov. 2011.

[6] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, "Chord: A Scalable Peer-to-Peer Lookup Service for Internet Applications," *ACM SIGCOMM CCR*, vol. 31, no. 4, 2001.

[7] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker, "A Scalable Content-Addressable Network," *ACM SIGCOMM CCR*, vol. 31, no. 4, 2001.

[8] D. Trossen and G. Parisis, "Designing and Realizing an Information-Centric Internet," *IEEE Mobile Communications*, vol. 50, no. 7, 2012.

[9] R. Alimi, L. Chen, D. Kutscher, H. H. Liu, A. Rahman, H. Song, Y. R. Yang, D. Zhang, and N. Zong, "An Open Content Delivery Infrastructure using Data Lockers," in *ACM SIGCOMM Workshop on Information-Centric Networking (ICN)*, 2012.

[10] C. Yi, A. Afanasyev, L. Wang, B. Zhang, and L. Zhang, "Adaptive Forwarding in Named Data Networking," *ACM SIGCOMM CCR*, vol. 42, no. 3, 2012.

[11] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, "Networking Named Content," in *ACM CoNEXT*, 2009.

[12] T. Koponen, M. Chawla, B.-G. Chun, A. Ermolinsky, K. H. Kim, S. Shenker, and I. Stoica, "A Data-Oriented (and Beyond) Network Architecture," in *ACM SIGCOMM*, Aug. 2007.

[13] P. Jokela, A. Zahemszky, C. Esteve Rothenberg, S. Arianfar, and P. Nikander, "LIPSIN: line speed publish/subscribe inter-networking," *ACM SIGCOMM CCR*, vol. 39, no. 4, 2009.

[14] S. Ratnasamy, B. Karp, S. Shenker, D. Estrin, R. Govindan, L. Yin, and F. Yu, "Data-Centric Storage in Sensornets with GHT, a Geographic Hash Table," *Mob. Netw. Appl.*, vol. 8, no. 4, 2003.

[15] A. Ghose, J. Grossklags, and J. Chuang, "Resilient Data-Centric Storage in Wireless Ad-Hoc Sensor Networks," in *Proc. of the International Conference on Mobile Data Management (MDM)*, 2003.

[16] T. N. Le, W. Yu, X. Bai, and D. Xuan, "A Dynamic Geographic Hash Table for Data-Centric Storage in Sensor Networks," in *IEEE Wireless Communications and Networking Conference (WCNC)*, vol. 4, 2006.

[17] M. Shorfuzzaman, P. Graham, and R. Eskicioglu, "Distributed Placement of Replicas in Hierarchical Data Grids with User and System QoS Constraints," in *IEEE 3PGCIC*, 2011.

[18] X. Tang and J. Xu, "QoS-Aware Replica Placement for Content Distribution," *IEEE Transactions on Parallel and Distributed Systems*, vol. 16, no. 10, 2005.