SLA-Driven Predictive Orchestration for Distributed Cloud-Based Mobile Services

Alexandru-Florian Antonescu*‡, Andre Gomes‡, Philip Robinson†, Torsten Braun‡
*SAP (Switzerland) Inc., Althardstrasse 80, 8105 Regensdorf, Switzerland
†SAP (UK) Limited, The Concourse, Queen's Road, Queen's Island, Titanic Quarter, Belfast BT3 9DT, UK
‡University of Bern, Communication and Distributed Systems (CDS), Neubrueckstrasse 10, 3012 Bern, Switzerland
alexandru-florian.antonescu@sap.com, gomes@iam.unibe.ch, philip.robinson@sap.com, braun@iam.unibe.ch

***Abstract*—We describe a system for performing SLA-driven management and orchestration of distributed infrastructures composed of services supporting mobile computing use cases. In particular, we focus on a Follow-Me Cloud scenario in which we consider mobile users accessing cloud-enable services. We combine a SLA-driven approach to infrastructure optimization, with forecast-based performance degradation preventive actions and pattern detection for supporting mobile cloud infrastructure management. We present our system's information model and architecture including the algorithmic support and the proposed scenarios for system evaluation.**

I. Introduction

The advent of cloud computing [1] has enabled organizations to better take advantage of the distributed architectures for their businesses. This includes benefiting from both the on-demand available computing, storage and network infrastructure resources and from the scalable distributed services architectures. However, this comes at the cost of increased complexity in managing such distributed computing environments, creating the need for dynamic adaptable infrastructure management systems based on formal defined models capable of optimizing the physical and virtual resource utilization.

Management of applications composed of multiple distributed services needs to take into consideration both resource load-balancing and the fact that a statically provisioned infrastructure might not yield the best results from both performance and costs point of view. For these reasons, both service instances and virtual computing, network and storage resources must be appropriately scaled in an automatic manner.

In case of cooperating cloud computing and mobile network providers, further application and infrastructure constrains appear due to the implicit user mobility and its implications on quality of experience perceived by the mobile users. This strengthens the requirement to dynamically reconfigure the infrastructure resources according to the varying workload. This also implies that the management system is capable of appropriately monitor the use of software and infrastructure resources and then of acting accordingly for preventing a service degradation beyond the acceptable limits.

A possible way of doing such convergent computing and network optimizations is using a cross-layer management system ([2], [3]). By using the distributed application's services guaranteed states (invariants), the system could take actions for preventing the deviation from the invariants. This allows taking an automated approach at both services and infrastructure dynamic reconfiguration based on machine learning techniques[4].

Using Service Level Agreements (SLA) management and data prediction for system reconfiguration enables both minimizing SLA violations caused by environment dynamics, and triggering infrastructure reconfiguration actions for optimizing resource utilization[5].

Cloud Computing has evolved to a more mature state in the last few years. While the main research focus has been at the datacenter and core network levels, it was assumed that edge networks and mobile devices have the capacity to deal with network congestions and data being accessed from different geographical locations. Although it was a right assumption some years ago, it is not anymore. The widespread usage of mobile devices, together with increasing application use of processing, storage and network resources, and new user requirements such as accessing their data from anywhere, have reduced the Quality of Experience (QoE) of the users and have created major issues for the Internet Service Providers around the world. Therefore, the need to extend the cloud core networks and datacenters arose. With it, challenges on how to distribute content and services between different types of datacenters (micro datacenters[1] and macro datacenters[2]), cache content in network equipment and how to provide content relocation on the fly also became topics desperately needing solutions from the research community.

In this paper, we aim to go even further and use mobility prediction to proactively decide when to move content and where to move it, so that SLAs are not violated and users are not harmful affected by their mobility or network conditions. This raises several questions, such as: which mechanisms to use for accurate macro mobility prediction, how to decide what has to be cached or moved or even how to make all the process transparent to the end user.

Given the limitations expressed by CAP (Consistency, Availability and Partition tolerance) theorem [6] regarding storage consistency and availability when network partitions

[1] Micro datacenters are medium to small-scale deployments of server clusters across a certain geographic area, which may cover locations ranging from a city to a small rural area as part of the mobile network infrastructure.

[2] Macro datacenters are the standard large-scale computing farms that are usually deployed and operated at a few strategically selected locations.

are present, we describe a SLA-driven architecture for maximizing the combination of consistency and availability in a mobile cloud context. We extend the work previously described in [5] and [7] by combining a machine learning approach to distributed resource provisioning with a SLA-driven approach at distributed services and resources orchestration.

This paper is structured as follows: Section II presents the related research topics, Section III describes the environment constraints and reference scenario in which our system operates, Section IV presents the system's components and their interactions, Section V discusses the proposed system evaluation, and finally, Section VI draws conclusions.

II. Background and Related Work

This section discusses relevant work in the areas of distributed cloud-based applications and approaches to infrastructure resource orchestration and management.

A. Distributed Cloud-based Applications

Cloud-based applications [1] are either proprietary software deployed on an Infrastructure as a Service (IaaS) cloud, applications built within a Platform as a Service (PaaS) or services offered by a Software as a Service (SaaS) provider. There are various motivations for adding distributed properties to cloud-based applications, including redundancy, load balancing, parallelization for performance increase or workflow, ownership and control constraints that force workloads to be placed in separate locations.

Distributed cloud-based applications consist of multiple network-connected workloads under the administration of a single user. Each workload may be assigned to a virtual machine, and each virtual machine might be exclusively or partially allocated a set of CPU cores, memory and storage in the cloud infrastructure. Furthermore, the workloads can be connected by application-level messaging using network overlays, virtual networks or physical ports configured to handle selected traffic. The distribution can be considered within a single or multiple datacenters, where different deployment, resource management and connectivity challenges arise.

Storage resources in cloud environments are typically accessed over the network, either directly or indirectly through a management system, such as databases or object storage systems. The cloud storage resources are usually distributed, fault tolerant, versioned and might be consistent, although eventually consistent [6] storage systems exists.

Various solutions exist for deployment specification and automation, such as described by Juve et al. [8] and Vecchiola et al. [9], but there is still a need for solutions to coordinating the scaling and resource usage optimization throughout the lifecycle of these complex cloud applications.

B. Mobile Cloud Computing

Mobile Cloud Computing [10] refers to the access of cloud computing-running services from mobile devices. Typical services being accessed from the mobile devices are multimedia services, along with enterprise services. The inherent mobility and limited computing resources of the mobile devices can leverage on the elastic cloud resources, as described further.

Follow-Me Cloud is an innovative concept that was presented by Bifulco et al. [11] as one way of increasing scalability of a mobile cloud management system. The key concept aims to support mobile users: a user moves and its data moves along to ensure the same QoE. However, this functionality involves complex questions that need to be tackled.

C. SLA Management in Cloud Environments

SLA management involves monitoring information about resource capabilities, availabilities and performance during operation, in order to execute actions for services to be delivered according to guarantees agreed with customers. There is then a need for mixing historical, predictive and live information about resources for dynamic re-planning and provisioning of resources. Further information includes the current load on different resources and the criticality of the service request to be handled. Given the on-demand model of cloud computing environments, there is an increased need for optimization of handling service requests and assigning resources based on multiple objectives. These multiple objectives are derived from the set of objectives in multiple SLAs and operational policies, such that conflicts and contentions will arise in an environment that allows concurrent services and service users.

Garg et al. [4] uses admission control and VM scheduling as their approach to SLA management. They use Artificial Neural Networks for predicting CPU utilization for making decisions about access control and starting/stopping VM instances.

Antonescu et al. [7] presents an architecture for performing cloud services deployment and scaling orchestration driven by SLAs together with with predefined infrastructure actions. We are extending that work in the context of mobile cloud with an inference and prediction engine capable of orchestrating actions for supporting Follow-Me Cloud concept.

Bobroff et al. [12] describes a system for dynamic placement of VMs considering resource utilization forecasting and SLA violation minimization by means of VM migration. Our works differs from their approach by extending the range of SLA-guaranteed infrastructure actions and introducing a more complex inferencing engine for predicting users mobility.

D. Outdoor Mobility Models

When it comes to mobility models and ways of representing outdoor mobility, many proposals exist in the literature. Bai et al. [13] provide a comprehensive overview of such proposals and classify them by types. This classification is made mainly by analyzing a set of key characteristics in detail:

- Random-based mobility models: the movement of a mobile node is random and without restrictions. Examples of these are the Random Waypoint Model, the Random Walk Model and the Random Direction Model. Each of them brings improvements by taking in account other factors and behaviors which are not human like.
- Models with Temporal Dependency: the movement of a mobile node is likely to be affected by its movement

history. Well know models are the Gauss-Markov Model and the Smooth Random Model. The second brings improvements over the first, as it also makes changes in speed and direction smoothly to prevent unrealistic movement behaviors.

- Models with Spatial Dependency: The mobile nodes tend to travel in a correlated manner. This can be achieved by using a reference point (leader node) for a group of moving nodes (Reference Point Group Model) or by having the nodes travel in a cooperative manner (Spatially Correlated Model).
- Models with Geographic Restriction: the movement of nodes is bounded by streets, freeways or obstacles. Known implementations are the Pathway Model and the Obstacle Model which restrict the movements by paths or by obstacles in the paths, respectively.

Even though these models are accurate enough to test many mobility mechanisms, to run our tests and gather real world valid conclusions of mobility prediction by learning we need to carry out our study with additional and more accurate mobility data. This more accurate data can be obtained using traces from real mobility, which will be presented later in this paper in Section V.

E. *Mobility Prediction with Artificial Intelligence*

There exist several techniques to determine prediction of a certain event on a timeline. These can be of two major types: statistical and artificial intelligence based. From the latter, recurrent neural networks are proven to be efficient when dealing with multi-step time series predictions, because they include internal dynamic memory, which grants a dynamic mapping of inputs to outputs.

These are the first assumptions of Kaaniche et al. [14] and Capka et al. [15], which aim to provide mobility prediction to optimize routing in Wireless Ad Hoc Networks and common Wireless Networks, respectively. They both use a recurrent and multi-layer neural network with three layers arranged in a feed forward fashion, together with a learning algorithm with an error function which is minimized using BPTT (Back Propagation Through Time) algorithm to train the network by adjusting the weights. Also, both their evaluations clearly shows that accurate prediction levels may be achieved if the architecture and learning algorithms are chosen correctly and the network is properly trained to a specific scenario. However, these approaches intend to improve routing, hence we need to adapt and improve it to deal with data relocation and resources allocation in a number of different mobile scenarios. Also, both obtain great simulation results when testing their implementations using mobility models such as the described before in this paper. But this does not validate the implementations for real world usage as we intend in this paper, which will require proper adaptions later described in Section IV.

III. Distributed Infrastructure and Application Optimization

The core concepts of our solution for distributed application orchestration, SLA management, provisioning, monitoring and infrastructure optimization are presented in this section.

A. *Reference Scenarios*

We consider a distributed computing environment composed of multiple pools of configurable computing resources (e.g. servers and storage) connected by high-speed network connections. The considered environment might spawn across multiple network domains.

The computing and storage resources are shared between regular users and mobile network operators. The mobile network operators consider multiple usages for the cloud infrastructure. They are providing to their users computational intensive services, such as support for secure communication services (e.g. secure messaging services), which can be accessed from the users' mobile terminals. These services are dynamically instantiated in various regions as the users demand for the secure services varies. This creates the basis of maintaining a balanced and cost effective distribution of services throughout the computing infrastructure.

Another scenario considered by the mobile network operators is offering access of high-quality, high-bandwidth multimedia services, such as video streaming or video conferencing. In case of video streaming, the storage and computing resources need to be carefully provisioned for ensuring that a proper level of quality of experience is maintained for the users base. This implies that based on access patterns, the video resources are replicated though the infrastructure for minimizing the network congestion and assuring an optimal access latency. Also, for the purpose of maintaining the quality of experience for users with mobile devices, on-the-fly video processing might be required (e.g. downscaling the video for matching various resolutions of the mobile users' devices). These two requirements imply that a careful orchestration of both storage and computing resources management needs to be done, while maintaining a guaranteed level for the quality of service and quality of experience for the users.

Considering that the mobile users are inherently accessing the network services while on the go, adds a new dimension to the infrastructure management requirements, of assuring that enough computing resources are made available for handling the users requests, without exceeding a maximal delay and without overloading the fixed network infrastructure connecting the mobile base stations to the actual computing centers.

In the context of the mentioned scenarios, it becomes obvious that a statically provisioned, tailor-made infrastructure, would lead to over-provisioning and sub-optimal use of resources, while increasing the energy costs associated with running the infrastructure. Expressing the quality of service guarantees in a measurable form, such as with Service Levels part of SLAs, would allow for a better visibility into the state of the network and computing infrastructure, while facilitating both a reactive and proactive elastic scaling of resources as the infrastructure utilization levels vary.

Having the basic blocks for observing the quality of service and experience throughout the infrastructure allows for detection of services and resources usage patterns, which, in turn, enable taking proactive actions for ensuring that the guaranteed service levels are not violated as a consequence of changing conditions in the service utilization levels. This also enables making energy savings due to a better sizing of active computing resources, combined with resource access planning. As TCP throughput depends on the round trip time [16] is also an important argument to deploying storage resources closer to the users.

IV. Proposed System Architecture

In this section we describe an extension to our existing SLA-based management system for distributed applications and infrastructure handling [7] for supporting storage resource management in the context of mobile network access. We present the components, their interactions and machine learning techniques used for forecasting mobile users locations and applications state in order to determine the appropriate actions to take for maintaining the SLA-invariants.

We define a new subsystem for handling the requirements of managing mobile accessed network storage resources, called Proactive SLA Infrastructure Manager. This new component, described in more detail in Section IV-B, is responsible for enabling proactive infrastructure management support. It contains the processing logic required for detecting user mobility patterns and for correlating these patterns with SLA-defined actions for maintaining the guaranteed quality of service and experience, as defined in the SLAs. Next, we present the information model required for supporting this subsystem.

A. Information model

We extend the SLA-centric information model described in [7] for modeling the dependencies between the physical and virtual resources' state, users' experience and SLA guaranteed states and dynamic actions.

The previously defined information model is centered on the SLA concept composed of guaranteed states and actions. The guaranteed states are expressions which refer to particular values of different metrics associated with resources and the services using those resources. The guaranteed actions are infrastructure reconfiguration actions, which should be executed when certain conditions called triggers become valid. The purpose of these actions is to prevent the invalidation of the SLA guaranteed states.

The information model also contains a view of the physical and virtual infrastructures. The physical infrastructure contains the relations between the physical computing, storage and network resources, while the virtual infrastructure contains the relations between the services running on the physical infrastructure. Both the physical and virtual resources are associated with monitoring metrics, which are then used for describing the SLAs' guaranteed states and actions' trigger expressions.

The actual state of the infrastructure resources is maintained as series of time and location bound monitoring metric values.

While this model is sufficient for driving a reactive control system, it is not enough for driving a proactive control system which learns which is the best infrastructure configuration state and network topology and then acts for ensuring that the identified configuration is reached. For supporting this feature we also define two components: a predicted resources' state component, which should hold the forecasted values of the monitoring metrics, and a predicted user mobility component that stores users past mobility and predicted future positions. This will enable scheduling of future actions based on the predicted resources state and SLA guaranteed states, so as based on the way the user moves along the network.

The predicted user mobility component should be fed with data from the previously described monitoring system, which needs to be converted into a format compatible with the prediction mechanisms and also divided by geographically divided zones in order to make more accurate predictions. We envision that the input format consists on a set of factors such as speed, heading, relative location in the world and likeliness to move (numeric, based on context. eg. user is at the airport). These together with the knowledge already in the dynamic memory of the predicting mechanisms and the division by geographical zones are in our opinion the needed parameters to determine a more accurate prediction, which in terms of output consists on a predicted location for the user. Moreover, using the error function of the mechanisms and establishing a level of confidence, we can define a threshold to decide if actions are needed or the prediction should be ignored.

Given the requirements of the 'follow-me-cloud' scenario, we also consider adding a new entity to the information model: the mobile users accessing the infrastructure provided services. This implies that network and possibly geographical location information needs to be associated with the users, for determining how the resources' location and network configuration affects the level of service experienced by the users. For fully supporting the user location awareness, the information model is also extended with time-bounded access patterns information and forecasted access patterns. Also, the Users model contains information about the values for the quality of service and quality of experience, from the mobile user's point of view.

Next, we show how this model is used for performing the infrastructure management functions.

B. Overall Component Architecture

In this section we give an overview of the important system components and their interactions, as depicted in Figure 1.

As described in [7], the SLA management system operates in a reactive manner, by processing the monitoring data on resources state and then selecting the appropriate actions for maintaining the SLA defined guaranteed states. We extend this mechanism with a proactive management subsystem (PSM)

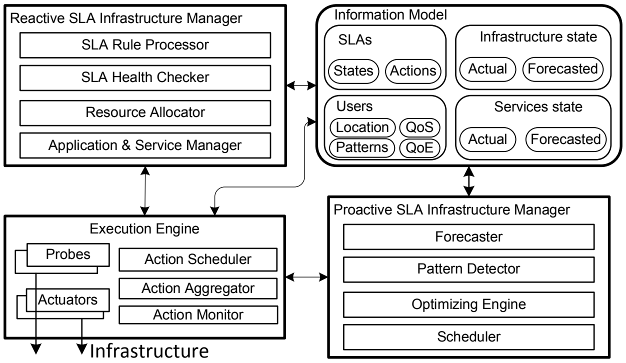

Fig. 1. System Model

for optimizing infrastructure configuration using forecasting and pattern detection mechanisms. The main components of the PSM subsystem are, as described in Fig. 1: the monitoring metric prediction engine (Forecaster), resource utilization Pattern Detector, Optimization Engine and Scheduler.

The system operates in an autonomous control loop, being driven by the SLA processing engines. The initialization is performed by registering the possible services along with their SLA profiles containing the SLA guaranteed actions, states and monitoring metrics in the Information Model. After the population of the infrastructure model by the Infrastructure Discovery component, the system is ready for receiving service deployment requests. Once a distributed application composed of different services has been deployed, the system begins the reactive infrastructure management process by identifying the suitable registered infrastructure actions at the SLA Rule Processor component, using the registered SLA.

The prediction engine provides forecasts of the resources' monitoring time series and user mobility. The Pattern Detector component processes the previous gathered resources' state along with its forecasts to extract utilization patterns in the network access of the services and storage resources. It also processes the users' real mobility and predicted mobility, in order to update the prediction mechanisms and make sure their predictions are as accurate as possible. The algorithms used by these components are described in Section IV-C.

The Optimization Engine then evaluates the forecasted metrics and utilization patterns against the SLA guaranteed states and actions' preconditions for determining whether certain corrective actions are required for preventing SLA violations.

The execution of the selected infrastructure optimization actions is then scheduled by the Scheduler component, taking into consideration the time horizon of the forecasts and the periods of the detected patterns.

The actual infrastructure actions are executed by the Execution Engine (EE) using information from the action templates database. The EE schedules the execution of the actions based on their dependencies using the Action Scheduler component. Multiple actions related to storage optimization are aggregated in the Action Aggregator component for ensuring that only significant changes are performed in the physical infrastructure. The actual execution is performed using the various Actuators, which are implementing different management APIs. Finally, the actions' execution is monitored though the Action Monitor.

The control loop is closed through the monitoring subsystem, which is recording the resources' state, used then for evaluating the SLAs at the SLA Rule Processor. The monitoring data along with the SLAs evaluation outcomes are recorded in the information model.

The system orchestrates multiple infrastructure optimization actions using three continuous phases throughout the lifetime of an application

1) **Analysis** gathering of monitoring data and evaluation of rules to determine if triggers have been reached and actions need to be performed for satisfying SLAs.
2) **Scheduling** determination of when determined actions need to be performed given the targets
3) **Execution** mapping the higher-level actions stated in rules to specific operations. This is more challenging in a heterogeneous environment as the specification of rules needs to be decoupled from the operational semantics of the underlying infrastructure.

C. Resource Utilization Prediction and Pattern Detection

The Resource Utilization Prediction component is responsible for forecasting the monitoring time series. It takes its input from the Monitoring component in the form of time series associated with each SLA metric of the registered services and infrastructure resources. The data is being processed by multiple prediction algorithms, such as Winter-Holts triple exponential forecasting, autoregressive integrated moving average (ARIMA) and exponential smoothing state space model with Box-Cox transformation, ARMA errors, Trend and Seasonal components (BATS). The actual forecasting algorithms execute in R Statistical Computing engine[17].

For detecting patterns in storage and network accesses the system will use statistical analysis of monitoring data, such as averaging over predefined time intervals (e.g. one hour), combined with forecasting. A small relative error of the forecasted time series with regards to the actual monitoring data would indicate that the observed variation represents a utilization pattern.

To predict user mobility, we intend to user neural networks based on work from the literature described in the Related Work section. These neural networks will eventually need validation, which will in fact happen after their definition and training. However, validation tests ran with a certain mobility model may be excellent and, at the same time, fail redundantly with another. And we also have to consider that these models are in fact very predictable, mostly because they lack of real world randomness and try to tackle a specific type of mobility while the world is composed of many types simultaneously. In our approach, we intend to validate our results by using real world traces that can be considered real world validation tests even though they are used for simulation purposes.

D. SLA-Driven Infrastructure Optimization

We are considering multiple possible infrastructure optimization actions. In the case of Follow-Me Cloud scenario, such actions are staring, migrating and stopping VM, as well as replication of storage resources.

In order to trigger such actions, information from multiple monitoring sources is used, in order to detect suboptimal performance of the infrastructure. Example of metrics supporting such decisions are the storage access rate, network access rates, latency and throughput, service waiting queue lengths, service processing times. These metrics would then be correlated with application and requests identifiers for creating user profiles, which would create the inputs for pattern detection components.

V. Discussion

For the system evaluation and validation we propose a three steps process, as described below. First, we will consider three scenarios in which multiple users are accessing storage resources, with and without requiring data processing (e.g. multimedia transcoding). Next, for each scenario we will generate utilization traces, simulating multiple user accesses. In parallel, we will gather actual monitoring traces from production systems and from small-scale real deployments. Using this combination of simulated and real-world data we will test the system scalability, stability and ability to adapt to varying infrastructure conditions.

We will train the neural networks both with mobility models, as described in Subsection II-D, and real-world traces, considering that the latter will also be used to validate the system (using a separate set). The challenge for validating the system with real-world data is obtaining a good representative set of human outdoor mobility traces. For this we will use vehicles mobility traces from taxi cabs monitoring data in San Francisco, USA [18]. On-foot human mobility traces will be obtained from five different sites [19]. With such representations of outdoor human mobility, we will complement the system's simulation, hence becoming closer to validate our proposal.

Finally, we envision a set of parameters that can be measured to evaluate our proposal. Namely, the prediction accuracy (%) gathered by comparing past predictions with real mobility data, the latency and jitter to access the content (ms) provided by the end users devices and also, for video streaming, two QoE metrics - Structural SIMilarity (SSIM) and Video Quality Metric (VQM). These last metrics can too be provided by the end users devices and feeded to the monitoring system.

VI. Conclusions

We described a system for performing SLA-driven management and optimizations of cloud computing infrastructures with a focus on supporting mobile cloud scenarios. We focused on a Follow-Me Cloud scenario involving optimization of distributed, network-connected storage resources and computing services. We combined a SLA-driven approach to infrastructure optimization, with forecast-based performance degradation preventive actions and pattern detection for supporting mobile cloud infrastructure management. We presented our system's information model and architecture including the algorithmic support and the proposed scenarios for system evaluation. Finally, we discussed about system validation plan and possible challenges for system evaluation.

Acknowledgments

The work in this paper has been (partially) funded by the European Union through projects GEYSERS (FP7-ICT-248657) and Mobile Cloud Networking (FP7-ICT-318109).

References

[1] P. Mell and T. Grance, "The nist definition of cloud computing (draft)," *NIST special publication*, vol. 800, p. 145, 2011.

[2] A.-F. Antonescu, P. Robinson, and M. Thoma, "Service level management convergence for future network enterprise platforms," in *Future Network & Mobile Summit (FutureNetw), 2012*, 2012, pp. 1–9.

[3] A.-F. Antonescu and P. Robinson, "Towards cross stratum sla management with the geysers architecture," in *Parallel and Distributed Processing with Applications, IEEE 10th Int. Symposium on*, 2012.

[4] S. Garg, S. Gopalaiyengar, and R. Buyya, "Sla-based resource provisioning for heterogeneous workloads in a virtualized cloud datacenter," *Algorithms and Architectures for Parallel Processing*, 2011.

[5] A.-F. Antonescu, P. Robinson, and T. Braun, "Dynamic sla management with forecasting using multi-objective optimizations," in *Integrated Network Management, IFIP/IEEE Symposium on*, 2013 (accepted).

[6] E. Brewer, "Towards robust distributed systems," in *ACM Symposium on Principles of Distributed Computing*, vol. 19, 2000, pp. 7–10.

[7] A.-F. Antonescu, P. Robinson, and T. Braun, "Dynamic topology orchestration for distributed cloud-based applications," in *Network Cloud Computing and Applications, IEEE 2nd Symposium on*, 2012.

[8] G. Juve and E. Deelman, "Automating application deployment in infrastructure clouds," in *Cloud Computing Technology and Science (CloudCom), IEEE Third International Conference on*, 2011.

[9] C. Vecchiola and R. Calheiros, "Deadline-driven provisioning of resources for scientific applications in hybrid clouds with aneka," *Future Generation Computer Systems*, pp. 58–65, 2012.

[10] D. Huang, "Mobile cloud computing," *IEEE COMSOC Multimedia Communications Technical Committee (MMTC) E-Letter*, 2011.

[11] R. Bifulco, M. Brunner, R. Canonico, P. Hasselmeyer, and F. Mir, "Scalability of a mobile cloud management system," in *1st MCC workshop on Mobile cloud computing*, 2012.

[12] N. Bobroff, A. Kochut, and K. Beaty, "Dynamic placement of virtual machines for managing sla violations," in *Integrated Network Management, 2007. IM '07. 10th IFIP/IEEE International Symposium on*, 2007.

[13] F. Bai and A. Helmy, "A survey of mobility modeling and analysis in wireless adhoc networks," in *Wireless Ad Hoc and Sensor Networks*. Springer, oct 2006, pp. 1–30.

[14] H. Kaaniche and F. Kamoun, "Mobility prediction in wireless ad hoc networks using neural networks," *J. of Telecommunications*, 2010.

[15] J. Capka and R. Boutaba, "Mobility prediction in wireless networks using neural networks," in *7th IFIP/IEEE International Conference on Management of Multimedia Netwoks and Services*. Springer Berlin Heidelberg, Oct. 2004, pp. 320–333.

[16] D. Milic and T. Braun, "Fisheye: Topology aware choice of peers for overlay networks," in *The 34th IEEE Conference on Local Computer Networks (LCN)*, 2009, pp. 467–474.

[17] R Development Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Austria, 2011. [Online]. Available: http://www.R-project.org

[18] M. Piorkowski, N. Sarafijanovoc-Djukic, and M. Grossglauser, "A Parsimonious Model of Mobile Partitioned Networks with Clustering," in *The First International Conference on COMmunication Systems and NETworkS (COMSNETS)*, 2009.

[19] I. Rhee, M. Shin, S. Hong, K. Lee, S. Kim, and S. Chong, "CRAWDAD data set ncsu (v. 2009-07-23)," http://crawdad.cs.dartmouth.edu/ncsu/mobilitymodels, 2009.