Issues in on-line performability modeling and evaluation through the model-based approach

Andrea Bondavalli
University of Florence, DSI Dept.
Viale Morgagni 65, I-50134, Firenze, Italy
bondavalli@unifi.it

Felicita di Giandomenico
CNR, ISTI Institute
Via Moruzzi 1, I-56124, Pisa, Italy
digiandomenico@isti.cnr.it

Abstract

Performability is widely recognized to be a highly representative metric of the dependability and Quality of Service of a system. The today complex networked systems show complex interdependencies with profound impact on the quality of services provided. In addition, adaptivity of the system architecture and configuration with respect to unforeseen changes that can occur at run-time becomes one of the most challenging aspects. This document describes the experience made by the authors on the dynamic management of performability driven reconfigurations, illustrates the authors vision of open problems and next challenges, and finally mentions the future work we will be doing in such context.

1. Introduction

Performability is widely recognized to be a highly representative metric of the dependability and Quality of Service of a system. Therefore, modeling and evaluation of performability are increasingly pursued as a major mean to assess the goodness of a system and/or to compare different architectural solutions at system design time. Today, information infrastructures are very complex networked systems where interdependencies among the several components play a relevant role in the quality of services they provide. In addition, in such system organization, adaptivity of the system architecture with respect to unforeseen changes that can occur at run-time becomes one of the most challenging aspects. Apart from natural system's evolution, many other sources of variability are possible, such as the occurrence of fault patterns different from those foreseen at design time, or the change of application's dependability requirements during the operational lifetime. To cope with unpredictability of events, approaches based on on-line system reconfiguration are necessary/desirable.

Such a dynamic framework poses new challenges also on the methods to analyze and evaluate dependability and QoS indicators, in order to identify the "best" reconfiguration in response to changes in system and/or environment conditions. In fact, to resort to off-line analysis and pre-plan reconfiguration strategies is practically feasible only in presence of a limited and well defined in advance number of situations requiring the application of a new reconfiguration policy in the system; i.e., for simple embedded systems. Unfortunately, especially for complex open systems, such a complete knowledge is not available, and situations may occur for which a satisfactory reaction has not been foreseen in advance. Therefore, the need of dynamically devising an appropriate answer to variations of the system and/or environment characteristics in order to achieve the desired level of performability and more generally dependability raises up. Consequently, on-line analysis is also required, to help realizing on-line system reconfiguration. Of course, although appealing, the online solution shows a number of challenging problems requiring substantial investigations.

In the next section, we shortly report on our experience in tackling the problem of dynamic system management, including dynamic reconfiguration supported by on-line performability, dependability and QoS assessment. Next, some identified challenges in the on-line analysis of systems through the model-based approach are briefly discussed.

2 Prior experience

Preliminary studies on dynamic system reconfiguration, assisted by on-line dependability and QoS assessment, have been carried on by our research group in the last few years. Our vision is to equip the system architecture with a "dependability manager" subsystem, which continuously supervises the controlled system and environment, ready to identify and apply reconfiguration policies at run-time. The architectural definition of such a general dependability manager needs to include an evaluation subsystem to provide quantitative comparison among several possible reconfiguration strategies. Model-based evaluation approaches are very suited to this purpose. Also, simple but effective performability indicators have to be defined, as a synthesis of a set of "alarming symptoms" detectable by specialized monitoring entities. When critical values are assumed by such performability indicators, a reconfiguration action is triggered. The model solution helps to devise the most appropriate configuration and behavior to face the actual situation. For example, through model solution the dependability of a new architecture of the system obtained by rearranging the remaining resources after a fault, or some performability indicator to carry out cost-benefit tradeoff choices can be evaluated.

Some early ideas on dynamic management of system resources have been developed in [3], where an optimal task admission policy for real-time fault tolerant systems has been proposed, based on a performability-like index, and its dynamic usage by the system planner is discussed. There, a "reflective" behavior of the planner is advocated, to allow computing the "optimal admission vector" on line, for systems subject to highly dynamic workload.

In [5], preliminary investigations have been conducted on a learning approach to perform database maintenance to dynamically adapt the maintenance policy at varying database and environmental parameter values leading to select, in each time period, the best maintenance policy available. In the proposed approach, an intelligent software agent acts as an audit manager, which learns to select an optimal action among defined relevant actions, by interacting with both an evaluation tool, used to predict a reward from a given state defined by a configuration of the relevant parameters, and the real environment, the behavior of which is sensitive to different configurations of the relevant parameters. Then, this intelligent software agent is trained on-line, using reinforcement Q-Learning, to learn the optimal maintenance policy to maximize the reward gained from the system.

In [10], a methodology and a framework for fault tolerance provision in distributed applications have been presented. The approach is based on monitoring the managed application to detect failures at component and operating system level, and on using dynamic reconfiguration for error recovery and/or for maintaining the system in a certain desirable state. The framework is enriched with a model based *Decision Maker*, which decides the most rewarding new reconfiguration for the managed application.

Therefore, how to reconfigure the system is decided at run time, following a set of pre-specified reconfiguration policies. The decision process is performed by using online evaluation of a stochastic dependability model which represents the whole system. Such modeling activity depends on the specified policy, on the requirements of the application and on the system status at reconfiguration time. In order to represent the topology of the managed application, which may change dynamically, the model is created at run time by assembling a set of sub–models (building blocks). Before the evaluation, these sub–models are opportunely instantiated to represent the single parts of the system (such as components, hosts and connectors) and initialized with information collected at run time. A simple example was also provided, to illustrate how the framework works. Models building blocks have been specifically set up for the components of the chosen system example, and how they are combined on-line and solved to represent the system reconfiguration strategy under evaluation is also shown. Of course, the composition of the building blocks is a very difficult and crucial aspect; it strictly depends on the specificity of the application at hand, and our example was simple enough to be managed. In general, the identification of appropriate compositional rules to dynamically build up the overall system model, of the criteria to base the reconfiguration selection process on and, primarily, of the necessary features to manage the complexity of the model solution in order to make the on-line evaluation a feasible approach, are in our view among the most challenging problems still under investigations.

In order to properly support our vision of dynamic modeling and solution of models for performability driven reconfigurations, model representation and model solution approaches needed to be enhanced. Our contribution to favor efficient on-line operation of stochastic evaluation can be found in [4], where hierarchical modeling has been widely discussed as a very appropriate approach to deal with dependability analysis of critical complex systems. In [9], a modeling methodology which exploits the power of the class of Markov regenerative stochastic Petri net models for phased-mission systems has been proposed. By exploiting the techniques available in the literature for the analysis of the Markov Regenerative Processes, an analytical solution technique with a low computational complexity was derived. These theoretical results have then been implemented in the tool DEEM, specialized for the dependability analysis of phased-mission systems [2].

Hierarchical model representation and model solution have been dealt with in [8], where a modeling methodology for hierarchical control systems has been proposed, and its application shown on a resource management system for wireless networks. By exploiting the characteristics of the specific, but well representative, addressed class of systems, the derived modeling methodology, mainly based on "abstract" and "detailed" models of the system components, is not only directed to build models in a compositional way, but it also includes some capabilities to reduce their solution complexity.

3 Major challenges in on-line modeling and evaluation

As already mentioned, most of the new challenges in dependability and QoS modelling and evaltuation are connected with the increasing complexity and dynamicity of the systems under analysis. Such complexity needs to be attacked both from the point of view of system representation and of the underlying model solution. On the basis of our previous experience and of some related papers in the literature (e.g., [7, 1, 6], just to mention a few), some issues and directions to go are discussed in the following.

A general modeling framework, not tied to a specific application but flexible enough to be easily adapted to different problems is a key challenge to address. In particular, a methodology has to be defined, allowing to identify systematically the input parameters of a performability driven dynamic manager, the metrics of interest and the criteria to base the decision on. These are very challenging issues, especially the last one, which is an instance of the well known and long studied problem of multiple-criteria decision making.

To promote efficiency in on-line modeling and solution, the basic idea is to build simplified (but still meaningful) models of the system to be controlled. The model simplicity in such a context is dictated by the need to solve the model dynamically as quickly as possible, in order to take appropriate decisions online. Too complex systems, in fact, would require too high computation time, thus defeating the effectiveness of the solution itself. In the following, some open issues are identified, which need to be adequately investigated in order to come out with satisfactory solutions to make on-line modeling and evaluation of dependability attributes (e.g., performability) affordable in practice.

3.1 State-space explosion and ways to cope with it

The state space explosion is a well known problem in model-based dependability analysis, which strongly limits the applicability of this method to large complex systems, or heavily impacts on the accuracy of the evaluation results when simplifying assumptions are made as a remedy to this problem. Modular and hierarchical approaches have been identified as effective directions; however, modularity of the modelling approach alone cannot be truly effective without a *modular solution* of the defined models.

Hierarchical approaches. Resorting to a hierarchical approach brings benefits under several aspects, among which: i) facilitating the construction of models; ii) speeding up their solution; iii) favoring scalability; iv) mastering complexity (by handling smaller models through hiding, at one hierarchical level, some modeling details of the lower one). Wide literature has been produced, adopting hierarchical methods to approach the analysis of complex systems ([1], just to mention one reference).

At each level, details of the architecture and of the status of lower level components are not meaningful, and only aggregated information should be used. Therefore, information of the detailed models at one level should be aggregated in an abstract model at a higher level. Important issues are how to abstract all the relevant information of one level to the upper one and how to compose the derived abstract models.

Composability. To be as general as possible, the overall model (at each level of the hierarchy) is achieved as the integration of small pieces of models (building blocks) to favour their composability. We define composability as the capability to select and assemble models of components in various combinations into a model of the whole system to satisfy specific application requirements. For the sake of model composability, we are pursuing the following goals:

- to have different building block models for the different types of components in the system. All these building blocks can be used as a pool of templates,
- to automatically instantiate an appropriate model, one for each component, from these templates, and
- at a given hierarchical level, to automatically link them together (by means of a set of rules which are application dependent), thus defining the overall model.

3.2 Multi-formalism integration

Different modeling techniques and models solution can be considered and integrated to pursue efficient on-line performability modeling and evaluation. In fact, efficiency is the topmost aspect to make the approach feasible on-line. Therefore, in accordance with a modular and hierarchical decomposition/composition of the system under analysis as already discussed in the previous subsection, the formalisms to be used to represent different modules and the relative solution techniques should be cautiously chosen so as to enhance, namely under the efficiency aspect, the overall modeling and evaluation process. E.g., simple but efficient formalisms, like combinatorial ones, should be privileged whenever possible; however, in more complex circumstances, like in presence of dependency relationships, state-space models are necessary. Integrating such multi-formalism adds further open issues to the composability aspects already introduced.

3.3 Dynamic solution

Models of components have to be derived online and combined to get the model of the whole system. Thus, compositional rules and the resulting complexity of the combined model solution appear to be the most critical problems to be properly tackled to promote the applicability of a dynamic approach to dependability (including performability) analysis. In particular, modularity of the model solution seems to be a paramount feature of a methodology to be suitable in dynamic contexts. In addition to hierarchical model solution already mentioned, the recently introduced “connectivity formalism” shows a very promising direction to limit state-space explosion and to promote highly efficient solution [6].

3.4 Assistance by automatic tools

Automatic tools are always a very powerful support to analyze systems, but, because of the high efficiency requirement, their usefulness becomes really paramount when on-line modeling and evaluation are accounted for. Current tools, implemented to be used to assist off-line dependability and QoS modeling and evaluation, have to be enhanced in order to be profitably employed on-line, by incorporating solutions that need to be found for the above mentioned open issues (and possibly others not yet identified so far). When a specialized system resource is dedicated to run the tool, this last could be kept continuously active in carrying on evaluations (e.g., of reconfiguration actions or other system aspects, according to the specific needs by the application at hand), and have the evaluation results maintained in a database, to be easily retrieved when of real utility during the system lifetime.

3.5 Current and future work

Dynamic management of large infrastructures and of applications that need to be supported by such frameworks require many challenging issues to be tackled with decision. A Dynamic Manager performing reconfigurations driven by a quantitative estimation of the performability achievable by systems in different settings is a desirable step towards a satisfactory solution for current and future systems.

Our contribution is meant in the same line as our past experience, where we will try to attack these problems through some challenging case studies that we have the fortune to be exposed to in EU-funded projects ready to start. One will deal with power grid distribution infrastructures; trying to understand their interdependencies and to figure out proper architectures and their correct management constitute the main objectives of the project. From the point of view of modeling and analysis, here a major difficulty which requires significant research investigations lies in the complexity of the modeled infrastructures in terms of largeness, multiplicity of interactions and types of the interdependencies involved. Another project will design critical applications supported by networking infrastructures where the usage of the different network supports and of the available resources has to be carefully and dynamically mixed. From the point of view of modeling and analysis, here the main challenges are posed by the fact that the environment and infrastructure are highly dynamic and show high inherent complexity due partly to their large-scale, but also to the variety of threats to be considered, to the heterogeneity of the design and to constant evolution.

References

[1] C. Betous-Almeida, and K. Kanoun. Stepwise construction and refinement of dependability models. In *Proc. IEEE International Conference on Dependable Systems and Networks DSN 2082*, Washington D.C., 2002.

[2] A. Bondavalli, S. Chiaradonna, F. Di Giandomenico, and I. Mura. Dependability modeling and evaluation of multiple-phased systems using DEEM . *IEEE Transactions on Reliability*, 53(4):509–522, 2004.

[3] A. Bondavalli, F. Di Giandomenico, and I. Mura. An optimal value-based admission policy and its reflective use in real-time dependable systems. *Real-Time Systems Journal, Kluwer Academic Publishers*, 16(1):5–30, 1999.

[4] A. Bondavalli, M. Nelli, L. Simoncini, and G. Mongardi. Hierarchical modelling of complex control systems: Dependability analysis of a railway interlocking. *Journal of Computer Systems Science and Engineering, CRL Publishing*, 16(4):249–261, 2001.

[5] A. Chohra, F. Di Giandomenico, S. Porcarelli, and A. Bondavalli. Towards optimal database maintenance in wireless communication systems. In N. Callaos, N. I. Da Silva, and J. Molero, editors, *The 5th World Multi-Conference on Systemics, Cybernetics and Informatics, ISAS-SCI 2001, Volume I: Information Systems Development*, pages 571–576, Orlando, Florida, USA, July 22-25 2001. IIIS.

[6] D. Daly and W. H. Sanders. A connection formalism for the solution of large and stiff models. In *34 Annual Simulation Symposium*, pages 258–265, Seattle, WA, 2001.

[7] B. R. Haverkort. Model-based self-configuration for quality of service. In *Proc. International Workshop SELF-STAR*, Bologna, IT, 2004.

[8] P. Lollini, A. Bondavalli, and F. Di Giandomenico. A modeling methodology for hierarchical control systems and its applications. *Journal of the Brazilian Computer Society, special Issue on Dependable Computing*, 10(3):57–69, 2005.

[9] I. Mura and A. Bondavalli. Markov regenerative stochastic Petri nets to model and evaluate the dependability of phased missions. *IEEE Transactions on Computers*, 50(12):1337–1351, 2001.

[10] S. Porcarelli, M. Castaldi, F. Di Giandomenico, A. Bondavalli, and P. Inverardi. A framework for reconfiguration-based fault-tolerance in distributed systems. In R. De Lemos, c. Gacek, and A. Romanovsky, editors, *Architecting Dependable Systems*, LNCS. Springer-Verlag, 2004. Also ICSE-WADS2003, Post-Proceeding of ICSE-WADS2003.