Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 28 (2014) 473 – 480

Procedia Computer Science

Conference on Systems Engineering Research (CSER 2014)

Eds.: Azad M. Madni, University of Southern California; Barry Boehm, University of Southern California; Michael Sievers, Jet Propulsion Laboratory; Marilee Wheaton, The Aerospace Corporation
Redondo Beach, CA, March 21-22, 2014

Emotions and the Engineering of Adaptiveness in Complex Systems

M.G. Sánchez-Escribano*, R. Sanz

Autonomous System Laboratory, Universidad Politécnica de Madrid, C/ José Gutiérrez Abascal 2, 28006, Madrid, Spain.

Abstract

A major challenge in the engineering of complex and critical systems is the management of change, both in the system and in its operational environment. Due to the growing of complexity in systems, new approaches on autonomy must be able to detect critical changes and avoid their progress towards undesirable states. We are searching for methods to build systems that can tune the adaptability protocols. New mechanisms that use system-wellness requirements to reduce the influence of the outer domain and transfer the control of uncertainty to the inner one.
Under the view of cognitive systems, biological emotion suggests a strategy to configure value-based systems to use semantic self-representations of the state. A method inspired by emotion theories to causally connect the inner domain of the system and its objectives of wellness, focusing on dynamically adapting the system to avoid the progress of critical states. This method shall endow the system with a transversal mechanism to monitor its inner processes, detecting critical states and managing its adaptivity in order to maintain the wellness goals. The paper describes the current vision produced by this work-in-progress.

Selection and peer-review under responsibility of the University of Southern California.

Keywords: Model-Driven Engineering, Runtime Model, Cognitive System, Emotional Models, Context-Awarenes

1. Introduction

Complex technical systems inevitably cause the emergence of complex engineering problems. Most of them are distributed systems-of-systems formed by different parts that cannot be functionally isolated but performing

* *E-mail address:* mguadalupe.sanchez@upm.es

1877-0509 © 2014 The Authors. Published by Elsevier B.V. Open access under CC BY-NC-ND license.
Selection and peer-review under responsibility of the University of Southern California.
doi:10.1016/j.procs.2014.03.058

federated work and sharing resources[1]. A major challenge of *systems engineering* is the building of dependable systems, regardless the operational environment or any unexpected events not predicted at design time. Complex systems can be regarded as *Open Systems* under the conception advanced by von Bertalanffy[2], i.e. "systems that maintain themselves in exchange of materials with the environment, and in continuous building up and breaking down of their components". This produces the emergence of *inner domains*, i.e. multiple states and service patterns that become a complex set of relevant circumstances that also influence the system. This concept of *domain* is a central aspect of the vision proposed in this paper. Furthermore, the external environment in which systems perform their missions is also harsh and full of unexpected events. Global uncertainty in unstructured environments and unpredictable internal and external events, cannot be fully considered in the design phases. At design time, the information about the operation of the system is partial, and the model for controlling behavior and acceptable changes is incomplete. However, the way in which complex systems deal with these unplanned events strongly influences its final resilience and robustness. This scenario reveals the need for designing novel methods to meet the requirements of complex systems, and autonomous self-organization is a key strategy for this end.

Ashby[3] explored several *Principles of the Self-Organizing System*, where "systems in general go to equilibrium" by the *Spontaneous Generation of Reorganization* among their multiple parts. Every transition from any state to one equilibrium state, requires the selection of proper states that determine the decisive stability. The challenge of building dependable complex systems should be addressed by designing systems with the capability to select those proper states, and this directly implies the design of systems with deep plasticity and self-awareness. The exploration of self-adaptive capabilities becomes a key matter and the *engineering of adaptiveness*, as the development of systematic methods to endow systems with adaptivity, becomes a critical need in complex systems engineering.

1.1. Models, adaptation and cognition background

How to encompass the influence of c*hange* wherever it comes from, is one of the main issues of software engineering for self-adaptive systems[4]. *Autonomic Computing* (AC)[5] is one of the new paradigms addressing those exigent needs from the viewpoint of self-adapting software. It deals with the self-management ability of distributed resources to attain system-level goals. *Self-adaptive software* aims to adjust different artifacts or attributes in response to changes in the *self* and in the *context* of a software system. The domain of *Self-X systems* addresses the architectural aspects of systems that are autonomous in the management of certain architectural traits[6,7]. The area of *Model-Integrated Computing* (MIC)[8] studies how to extend the scope and use of models to maintain a correlated sequence of changes between the model itself and the complex system in which it is embedded. A main feature of MIC is that it tries to match modeling paradigms to the needs of the systems engineering domain. *Model Driven Software Development* (MDSD)[9,10] focuses on using software models for increasing the quality and improving the effectiveness of the software process. A more open-domain line of research is what is referred as *Model Driven Engineering* (MDE)[11]. Within this area, *models@runtime* [12, 13] creates models used to reason about the system and its operating environment at run time[14]. Lastly, the dynamic identification of *changing requirements* is one of the new research challenges to assure the reliability of systems under unplanned or unexpected occurrences[4].

The approach suggested in this paper addresses the issue from the analysis of the cognitive architecture of biological emotion. There are plenty of examples of cognitive architectures that address partial aspects of human cognition[15,16]. From a more systems centric view, examples go from studies about *end-means patterns* as hierarchical organizations for structuring goals and means, to a large number of applicable analyses about complexity in engineering systems[3,17], organized systems[18], risk management[19] or entropy growth management, i.e. adjustment of structural artifacts to respond to the challenges of the environment[20]. The focus on emotion is not about the perceivable aspects of it[21] but about the structural traits concerning the adaptation of system structure to shape behavior[22]. All of this conforms a wide field of fuzzy subfields. We will focus on those paradigms of relevance for this paper such as self-adaptiveness, model centric and cognitive architectures.