Evolution of Verification Techniques by Increasing Autonomy of Cooperating Agents

Francesca Saglietti, Sven Söhnlein, and Raimar Lill

Abstract. As system parts are becoming increasingly decoupled, gaining at the same time in terms of local autonomy, this article elaborates on the effects this trend has on verification and validation techniques. Both qualitative approaches to fault detection and quantitative approaches to reliability assessment are analyzed in the light of their evolution to adapt to the increasing decentralization and autonomy of modern ‘systems of systems’.

1 Introduction

The responsibility role assumed by software-based applications when controlling a variety of critical tasks is continuously increasing. While in previous times the limited functionality of such systems allowed for monolithic designs, their increasing complexity has led - over the past decades – to growing levels of modularity, ranging from simple well-structured programs to sophisticated component-based systems including pre-developed packages.

Evidently, the structure introduced by modularity strongly supports transparency and understandability by endorsing the principles of abstraction and of separation of concerns, in particular contributing to lean maintenance by favoring change management via highly cohesive modules (easing fault localization) and loosely coupled modules (reducing potential side effects arising during fault removal). On the other hand, the insertion of pre-developed components was shown to introduce new fault sources in case the original application context in which the components originate slightly differs from the future context addressed by the new system to be developed [5].

This trend towards decoupled software parts is further increasing, typically for systems whose functional scope is dynamically evolving with time, like

- critical services provided on internet platforms (cloud computing);
- controllers protecting different assets considered as essential for the functioning of a society (critical infrastructures), like energy generation, transmission

Francesca Saglietti · Sven Söhnlein · Raimar Lill
Chair of Software Engineering
University of Erlangen-Nuremberg, Erlangen (Germany)
e-mail: {saglietti,soehnlein}@informatik.uni-erlangen.de
and distribution, telecommunication, water supply, food production and distribution, public health, transportation, financial and security services;
- mechanical, intelligent agents performing individual or common tasks (cooperative robots).

While the motivation leading to such decentralized, highly decoupled and autonomous sub-systems – namely, ease of system evolution over time as well as growth of time efficiency and service flexibility – clearly strengthens their attractiveness, the high dependability demands posed on such software applications render their verification even harder than is already the case for more modest functional scopes.

In fact, as the individual system parts evolve with time and behave at a high degree of autonomy, the multiplicity of their interplay increases at rapid pace and is extremely difficult to be systematically captured by testing. On the other hand, in order to verify acceptable behavior, an accurate preliminary analysis should identify the variety of potential scenarios involving the interaction of autonomous parts and to assess their adequacy by representative test cases.

The intention of the present contribution is to elaborate on this novel challenge posed to software reliability engineering by illustrating recent and ongoing work addressing the evolution of verification activities with increasing autonomy of system parts.

2 Autonomy

The ancient Greek root of the term ‘autonomy’ (auto = self + nomos = law) reveals that its original meaning generally referred to entities providing themselves with their own laws. Depending on the underlying political or ethical context, this definition may allow for slightly diverse interpretations.

- within a political context, it refers to the self-government of human populations, while
- in terms of moral philosophy, it refers to the moral responsibility of an individual for his / her actions.

In both cases, it involves the capacity of an individual entity (human, population, or technical system) to make a rational and informed decision. In spite of the idealistic content hidden behind this assumption, it is well-known to software engineers – as it is to politicians and philosophers – that full autonomy (i.e. completely decoupled individual decisions) cannot help achieving an overall prioritized target of a society or an application.

The opposite, namely central controllers fully dictating actions to agents, evidently severely limit the potential capabilities provided at local level, restricting both performance and flexibility.