Chapter 7
On Solving the Multi-Objective Software Package Upgradability Problem

Noureddine Aribi
Lab. LITIO, University of Oran 1, Oran, Algeria

Yahia Lebbah
Lab. LITIO, University of Oran 1, Oran, Algeria

ABSTRACT

Free and open source software (FOSS) distributions are increasingly based on the abstraction of packages to manage and accommodate new features before and after the deployment stage. However, due to inter-package dependencies, package upgrade entails challenging shortcomings of deployment and management of complex software systems, inhibiting their ability to cope with frequent upgrade failures. Moreover, the upgrade process may be achieved according to some criteria (maximize the stability, minimize outdated packages, etc.). This problem is actually a multi-objective optimization problem. Throughout the article, the authors propose a Leximax approach based on mixed integer linear programming (MILP) to tackle the upgradability problem, while ensuring efficiency and fairness requirements between the objective functions. Experiments performed on real-world instances, from the MANCOOSI project, show that the authors' approach efficiently finds solutions of consistently high quality.

1. INTRODUCTION

Free and Open Source Software (FOSS) distributions (Di Cosmo, Zacchiroli, & Trezentos, 2008) are among the most complex software systems known, being made of tens of thousands deployment units known as packages. These packages evolve rapidly and are developed and released independently without a priori coordination or central authority able to control the involved parties (Di Cosmo et al., 2008; Michlmayr, Hunt, & Probert, 2007). Owing to this situation, difficult issues are raised for both software editors and system administrators. For instance, system upgrade in a GNU/Linux distribution

DOI: 10.4018/978-1-7998-9158-1.ch007

may proceed on different paths and requires the presence of a set of previously installed packages, and the absence of another set of packages. Hence, in some cases, it is not possible to install or upgrade all the desired packages and possible failures can occur during upgrades.

Research works (cf., (Argelich & Lynce, 2008)) developed in the context of the MANCOOSI [1] (Managing the Complexity of the Open Source Infrastructure) project, aim at developing tools for the system administrator in order to handle complex inter-package dependencies and frequently available package upgrades. These tools are used regularly to address security issues, bug fixes or to add new features that respect user's preferences. Besides, the predecessor EDOS project (Treinen & Zacchiroli, 2008b) had focused on tools for the distribution editor. When performing packages upgrade, user's preferences are expected to be handled in a consistent and efficient way, which is a current hot topic in Artificial Intelligence with active research lines in constraint satisfaction and optimization (Junker, 2004; Rossi, Venable, & Walsh, 2008). When upgrading packages, for instance, one can choose to minimize the whole size of the packages to install, to minimize the number of packages to install, to install the recent versions of packages. All these criteria are objective functions, subject to constraints stating the dependency and avoiding conflicts. By this way, upgrading packages comes back to a multi-objective optimization problem, which has been tackled by the MANCOOSI project. In optimization problems with a single criterion, the goal is to find an optimal solution such that the objective function is minimized or maximized. Hence, the problem is said to be well defined. Alternatively, if the optimization problem comprises more than one criterion (which is the case of most real-world optimization problems), then the solution of the problem becomes difficult to characterize. In fact, we cannot find (in general) a feasible solution that optimizes the whole criteria at the same time. Therefore, what we need is an efficient ordering method that aggregates all these objective functions (aka criteria) into a single and global objective function, which is the strategy, adopted in most optimization methods (Argelich & Lynce, 2008; Argelich, Lynce, & Silva, 2009).

Mixed Integer Programming (MIP) (Wolsey, 1998) is one of the most important techniques for solving complex optimization problems. In this paper, the authors propose a linear *Leximax* approach tailored for solving the upgradability problem, while ensuring efficiency and fairness (equity) requirements. The main intuition behind the equity criterion refers to the idea of selecting solutions that fairly share satisfaction between objective functions (Sen and Foster, 1997). The proposed approach is modeled as a MIP model. Next, this approach is applied on the package upgradability problem (using benchmarks coming from the MANCOOSI project), where the criteria and constraints are linear on boolean domains.

The rest of this paper is organized as follows. Section 2 provides some necessary preliminaries. Section 3 describes the package upgradabilily problem. Section 4 illustrates our approach on a practical example. Section 5 highlights the contribution of this paper. Sections 6 describe with details the proposed approach. Experimental results are given and discussed in Section 7. We present some related works in Section 8. Section 9 concludes the paper.

2. PRELIMINARIES

Before discussing our approach, we provide some necessary background.