Real World System Architecture Design Using Multi-criteria Optimization: A Case Study

Rajesh Kudikala\textsuperscript{1}, Andrew R. Mills\textsuperscript{1}, Peter J. Fleming\textsuperscript{1}, Graham F. Tanner\textsuperscript{2}, and Jonathan E. Holt\textsuperscript{2}

\textsuperscript{1} Department of Automatic Control and Systems Engineering, The University of Sheffield, Mappin Street, S1 3JD, Sheffield, UK
\{r.kudikala,a.r.mills,p.fleming\}@sheffield.ac.uk
\textsuperscript{2} Rolls-Royce plc, PO Box 31, Derby, UK
\{graham.tanner,jonathan.holt\}@rolls-royce.com

Abstract. System architecture design using multi-criteria optimization is demonstrated using a case study of an aero engine health management (EHM) system. A design process for optimal deployment of EHM system functional operations over physical architecture component locations, e.g., on-engine, on-aircraft and on-ground, is described. The EHM system architecture design needs to be optimized with respect to many qualitative criteria in terms of operational attributes within the constraints of resource limitations. In this paper the system architecture design problem is formulated as a multi-criteria optimization problem. Considering the large discrete search space of decision variables and many-objective functions and constraints, an evolutionary multi-objective genetic algorithm along with a progressive preference articulation technique, is used for solving the optimization problem. The optimization algorithm found a family of Pareto solutions which provided valuable insight into design trade-offs. Using the progressive preference articulation technique, the optimization search can be focused for the industrial decision maker on to a region of interest in the objective space. Performance of the proposed method is evaluated using various test metrics. Using this approach it was possible to identify the most significant design constraints ("hot spots") and the opportunities afforded by either the relaxation or the tightening of these constraints, along with their performance implications.

Keywords: System architecture design, multi-criteria optimization, many-objective optimization, preference articulation, genetic algorithms.

1 Introduction

The architecture of a complex system can be described in terms of functional requirements, physical elements, and element interrelationships. It is often seen as a generic framework or blueprint of the overall system. Designing a complex system architecture can be a difficult task involving multi-faceted trade-off decisions. The design process often needs to consider experience, models and
data from many design disciplines. A typical architecture design process starts by identifying the main functional requirements and follows a process of decomposition. The top level system functional requirements (use cases) are divided into several sub-functions. The physical form of the system is also divided into sub-systems and components. Designers try to map the functionalities onto the physical hardware components. Designers then iterate between the upper and lower levels of the system decomposition to optimize the deployment of functional operations onto physical systems. However, due to the large and discontinuous design search space, many qualitative and quantitative criteria, it is almost impossible for designers to find optimal architecture designs. Designers have considered various methods and tools for exploring these trade-offs in the design space.

Since they are efficient, global, parallel search methods, evolutionary algorithms (EAs) are very popular for solving such complex design problems. These algorithms apply generic operators of variation and thus are applicable to search non-standard combinatorial search spaces taking into account domain-specific characteristics of the problem. Thompson et al. explored the potential of the increasing use of embedded intelligence through deployment of smart sensors and actuators in future distributed control system (DCS) architectures for aero-engines. They have used a multi-objective evolutionary algorithm (MOEA) to generate and assess competing architectures. A model-driven architecture design approach, SESAME, developed in, which explores the design space of embedded system architectures using an MOEA. Similarly, the Palladio component modelling (PCM) framework was developed to automatically evaluate software architectures with an MOEA. Armstrong et al. developed a tool set for the function based architecture design exploration, which included function decomposition, adaptive function mapping and complex inter relations between architecture elements. This information aids in architectural definition and trading of architecture alternatives. Other frameworks adopted multiple stages and commercial life-cycle viewpoints in terms of qualitative and quantitative analysis for evaluating the architecture trade-off decisions.

Many of the architecture design frameworks concentrated on only multi-objective problems having 2 to 3 objective functions. However, in general real-world design problems will have many-objective functions to be optimized. Fonseca and Fleming introduced the first Pareto-optimal MOEA; it was named MOGA - multi-objective genetic algorithm. There are many different MOEAs available now for solving multi-objective optimization problems such as, NSGAII, SPEA2 etc., However, their search ability can significantly deteriorate when solving many-objective optimization problems. The optimal trade-off surface of a multi-objective optimization problem can contain a potentially infinite number of Pareto-optimal solutions. The task of an MOEA is to provide an accurate and useful representation of the trade-off surface to the decision-maker (DM). With the increased number of objectives, the number of non-dominant solutions increases exponentially, decreasing the selection pressure and convergence towards true Pareto optimal surface. Visualization of the