ARTICLE IN PRESS
SCICO:2149

Science of Computer Programming ••• (••••) •••–•••

Contents lists available at ScienceDirect

Science of Computer Programming

www.elsevier.com/locate/scico

Architecture-level configuration of industrial control systems: Foundations for an efficient approach ☆

Razieh Behjati [a,*], Shiva Nejati [b]

[a] Simula Research Laboratory, Norway
[b] SnT Centre, University of Luxembourg, Luxembourg

ARTICLE INFO

Article history:
Received 16 April 2016
Received in revised form 30 September 2017
Accepted 6 October 2017
Available online xxxx

Keywords:
Model-based configuration
CSP
Backtracking
UML/OCL

ABSTRACT

Configuration is a recurring problem in many domains. In an earlier work, we focused on architecture-level configuration of large-scale embedded software systems, in particular industrial control systems, and proposed a methodology that enables engineers to configure products by instantiating a given reference architecture model. Products have to satisfy a number of constraints specified in the reference architecture model. If not, the engineers have to backtrack their configuration decisions to rebuild a configured product that satisfies the constraints. Backtracking configuration decisions makes the configuration process considerably slow. In this paper, we improve our earlier work and propose a backtrack-free configuration mechanism. Specifically, we propose an algorithm that computes an ordering over configuration parameters that, for any cycle-free reference architecture model, yields a consistent configuration without any need to backtrack. We provide formal specification and proofs of termination, correctness, and completeness of our algorithm. We demonstrate the effectiveness of our approach using a simplified industrial case study. Results of our experiments show that our ordering approach eliminates backtracking in practice. It reduces the overall configuration time by reducing both the required number of value assignments, and the time that it takes to complete one configuration iteration. Moreover, we show that the latter has a linear growth with the size of the configuration problem.

1. Introduction

Configuration is a recurring problem in many embedded software system domains such as energy, automotive, and avionics, where product-line engineering approaches [29,25] are applied to support configurability. The class of embedded software systems in these domains is referred to as industrial control systems or integrated control systems (ICSs) [4]. In our earlier work [4], we performed a comprehensive domain analysis and identified common characteristics of ICS families, and their configuration challenges. In particular, these systems are highly configurable and involve mechanical and electrical components in addition to software. The software components need to be configured so that they can interact with and

☆ This paper is an extension of a conference paper titled "Efficient Architecture-Level Configuration of Large-Scale Embedded Software Systems" published in the proceedings of the Sixth International Conference on Fundamentals of Software Engineering (FSEN), 2015 [5].

* Corresponding author.
E-mail address: behjati@simula.no (R. Behjati).

https://doi.org/10.1016/j.scico.2017.10.001
0167-6423/

Please cite this article in press as: R. Behjati, S. Nejati, Architecture-level configuration of industrial control systems: Foundations for an efficient approach, Sci. Comput. Program. (2017), https://doi.org/10.1016/j.scico.2017.10.001

control different hardware component configurations. To address these challenges, we proposed the SimPL methodology [4] for modeling *reference architectures* of ICS families, and a configuration approach [3] for developing individual systems (i.e., configurations) from such reference architectures.

Briefly, a reference architecture provides a common, high-level, and customizable structure for all members of the product family [29] by specifying different types of components and configurable parameters, as well as, constraints capturing relationships between these parameters. Through configuration, engineers develop each product by creating component instances and assigning values to their parameters such that the constraints are satisfied.

Normally, configuring industrial control systems involves assigning values to tens of thousands of interdependent parameters [4]. Typically, 15 to 20 percent of these parameters are interdependent. Finding consistent values for interdependent parameters without any automated support is challenging. Manual configuration – the common practice in many companies – is time-consuming and error-prone, especially for large-scale systems. During the last three decades, researchers have developed a large number of approaches to automate various configuration use cases [7]. Most of these approaches concern consistency of configuration decisions, and rely on constraint solvers (e.g., [15,26,21]) or SAT solvers (e.g., [22]) for ensuring consistency.

In our earlier work [3], we proposed an iterative approach for configuring industrial control systems where at each iteration the value for one parameter is specified. If at some point during the configuration, a value assignment violates some constraints, then the engineers may have to *backtrack* some of their recent choices until they can find a configuration assignment consistent with the constraints in the reference architecture. Backtracking configuration decisions makes the configuration process considerably expensive.

In this paper, we propose a new approach that eliminates backtracking during configuration by configuring parameters in a certain order. The core idea of our approach was presented in [5]. The present article extends and refines our earlier work, making the following contributions:

- We explain how an ordering is extracted from a cycle-free[1] reference architecture model.
- We provide formal specifications, and prove that if the ordering is followed, our algorithm generates consistent and complete configured products without any need to backtrack a decision.
- We argue that elimination of backtracking considerably improves the performance of our configuration approach. We show this by applying our approach to a simplified excerpt of an industrial case study from the oil and gas domain. The experiment shows that our ordering approach reduces the overall configuration time by both reducing the required number of value assignments, and reducing the time that it takes to complete one configuration iteration. Further, we demonstrate that in our backtrack-free configuration approach the time required for completing one configuration iteration grows linearly with the size of the configuration problem. In our original configuration approach, this time has a quadratic growth.

In the rest of the paper, we first present the related work and position our work in the literature. Sections 3 and 4 provide the background on constraint solving techniques, and product family modeling and configuration, respectively. In Section 5, we describe the problem of backtracking configuration decisions. A formal specification of the main concepts in consistent configuration is presented in Section 6. We utilize this formal specification in the description of our ordering approach as well as the proofs. Our ordering approach for eliminating backtracking, and the resulting backtrack-free configuration algorithm are presented in Section 7. Termination, consistency, and completeness properties of our configuration approach are stated and proved in Section 8. In Section 9, we experimentally evaluate the efficiency and scalability of our approach. The applicability of our approach, and future work are discussed in Section 11. Finally, we conclude the work in Section 12.

2. Related work

Existing configuration approaches fall into two general categories, *non-interactive* and *interactive*. Most configuration approaches belong to the first category, where the objective is to produce some final configured products without requiring intermediate input from users. They may either find an optimized solution based on some given optimization criteria (e.g., [15,21]) or find all configuration solutions (e.g., [8,12,26]). The non-interactive approaches may either rely on meta-heuristic search approaches [16,18,23], or on systematic search techniques used in constraint solvers [10,11,20], or on symbolic decision procedures [9]. Among these, meta-heuristic search approaches are generally faster and require less memory. However, since meta-heuristic search is stochastic and incomplete, it cannot support an interactive process where engineers have to be provided with precise and complete guidance information at each iteration.

Interactive configuration methods (e.g., [19,22,33,34]) mostly rely on constraint solvers or symbolic reasoning approaches. Backtracking is required whenever an inconsistency arises, even though it may make the process considerably slower. In general, constraint solvers alleviate the drawbacks of backtracking by employing heuristics such as back-jumping [13], iden-

[1] In our approach, a reference architecture model consists of a UML class model, including a decomposition hierarchy, and a set of constraints (see Section 4.1). Such a reference architecture is cycle-free if it neither contains any cycles in the decomposition hierarchy nor in the constraints. This condition is required to ensure the termination of the configuration process as well as the complete elimination of backtracking (see Sections 8 and 11).

Please cite this article in press as: R. Behjati, S. Nejati, Architecture-level configuration of industrial control systems: Foundations for an efficient approach, Sci. Comput. Program. (2017), https://doi.org/10.1016/j.scico.2017.10.001