Extended Abstract: ProDSPL: Proactive Self-Adaptation based on Dynamic Software Product Lines

Inmaculada Ayala
Universidad de Málaga
Málaga, Spain
ayala@lcc.uma.es

Alessandro V. Papadopoulos
Mälardalen University
Västerås, Sweden
alessandro.papadopoulos@mdh.se

Mercedes Amor
Universidad de Málaga
Málaga, Spain
pinilla@lcc.uma.es

Lidia Fuentes
Universidad de Málaga
Málaga, Spain
lff@lcc.uma.es

ABSTRACT

This is an extended abstract of the article: Inmaculada Ayala, Alessandro V. Papadopoulos, Mercedes Amor, Lidia Fuentes, ProDSPL: Proactive self-adaptation based on Dynamic Software Product Lines, Journal of Systems and Software, Volume 175, 2021, 110909, ISSN 0164-1212, https://doi.org/10.1016/j.jss.2021.110909.

CCS CONCEPTS

• **Software and its engineering → Software product lines**; • **Computer systems organization → Reliability**; **Reconfigurable computing**.

KEYWORDS

Dynamic Software Product Lines, Proactive Control, Self-Adaptation, Optimization, Linear constraint

ACM Reference Format:
Inmaculada Ayala, Alessandro V. Papadopoulos, Mercedes Amor, and Lidia Fuentes. 2021. Extended Abstract: ProDSPL: Proactive Self-Adaptation based on Dynamic Software Product Lines. In *25th ACM International Systems and Software Product Line Conference - Volume A (SPLC '21), September 6–11, 2021, Leicester, United Kingdom.* ACM, New York, NY, USA, **??** pages. https://doi.org/10.1145/3461001.3473064

Dynamic Software Product Lines (DSPLs) are a well-accepted approach to self-adaptation at runtime. In the context of DSPLs, there are plenty of reactive approaches that apply countermeasures in the form of a new system configuration as soon as a context change happens. As new configurations are usually generated at runtime for a specific change, these reactive strategies can be resource-consuming tasks that can negatively affect battery-powered devices' performance. Therefore, it is important to consider the adaptation cost when we are opting for a DSPL or another.

In some scenarios, the adaptation process can take a significant amount of resources of the system, or it is not acceptable to stop–or delay–the system's normal functioning for a reconfiguration. These scenarios require techniques that lessen the number of reconfigurations but maintain a good quality of service at the same time or, in other words, they require a *proactive strategy*. Proactive strategies consider the current context and the system's expected evolution over time, so their adaptation solutions tend to stay valid for longer. But the price to pay when we apply a proactive strategy is that sometimes the calculated application configuration has a lower quality than the solution provided by some reactive approaches. Therefore, the great challenge is finding a proactive strategy whose adaptation solutions are good enough to increase the system's lifespan while reducing required reconfigurations.

We propose ProDSPL, a proactive DSPL approach that exploits an automatically learnt model of the system to anticipate its future variations and generates the best DSPL configuration to lessen the negative impact of future events. Predicting the future fosters good adaptations for a longer time and reduces the number of reconfigurations required. ProDSPL combines DSPL with a control-based proactive decision strategy to generate optimal configurations. We formulate the DSPL reconfiguration service as a single-objective optimization problem over a prediction horizon subject to the linear constraints of the DSPL feature model. This formulation is based on a mapping between extended feature models and linear constraints, which is part of our solution to combine proactive control with DSPL. Although previous works have proposed this kind of mapping, they only consider basic feature models.

We validate ProDSPL by comparing the performance and the quality of its results with DAGAME, a reactive approach that uses a DSPL approach with a genetic algorithm that generates configurations at runtime. Considering the concept of optimality, DAGAME can obtain solutions with optimality higher than 87.4% and execution times between 20 and 100 milliseconds. In this paper, ProDSPL performance is evaluated using a simulator of a mobile app developed in Java to make experiments and results reproducible by third parties. Compared with the purely reactive approach, ProDSPL shows good results concerning the performance and the quality of the configurations generated. Furthermore, the results obtained support in practice our initial hypothesis about the use of a proactive approach. That is, **our solution is more sustainable as it contributes to reducing the overall cost–in time and energy–of reconfiguration and leads to more stable systems while maintaining a satisfactory quality of adaptation**.

SPLC '21, September 6–11, 2021, Leicester, United Kingdom

ACM ISBN 978-1-4503-8469-8/21/09.
https://doi.org/10.1145/3461001.3473064