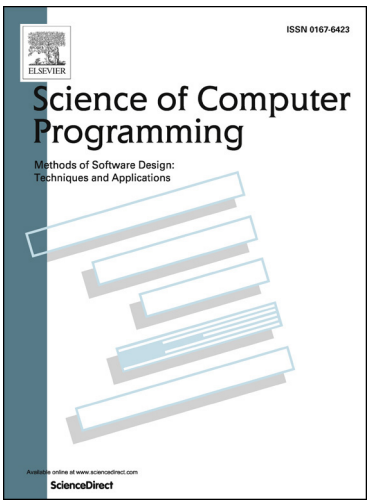
Accepted Manuscript



Reasoning about sensing uncertainty and its reduction in decision-making for self-adaptation

Javier Cámara, Wenxin Peng, David Garlan, Bradley Schmerl

PII: S0167-6423(18)30252-1
DOI: https://doi.org/10.1016/j.scico.2018.07.002
Reference: SCICO 2222

To appear in: *Science of Computer Programming*

Received date: 29 December 2017
Revised date: 27 June 2018
Accepted date: 3 July 2018

Please cite this article in press as: J. Cámara et al., Reasoning about sensing uncertainty and its reduction in decision-making for self-adaptation, *Sci. Comput. Program.* (2018), https://doi.org/10.1016/j.scico.2018.07.002

Reasoning about Sensing Uncertainty and its Reduction in Decision-Making for Self-Adaptation

Javier Cámara[a], Wenxin Peng[b], David Garlan[a], Bradley Schmerl[a]

[a]*Carnegie Mellon University. 5000 Forbes Ave. Pittsburgh, PA 15213, USA*
{jcmoreno,garlan,schmerl}@cs.cmu.edu
[b]*Google Inc. 1600 Amphitheatre Parkway Mountain View, CA 94043, USA*
wenxinp@google.com

Abstract

Adaptive systems are expected to adapt to unanticipated run-time events using imperfect information about themselves, their environment, and goals. This entails handling the effects of uncertainties in decision-making, which are not always considered as a first-class concern. This paper contributes a formal analysis technique that explicitly considers uncertainty in sensing when reasoning about the best way to adapt, together with uncertainty reduction mechanisms to improve system utility. We illustrate our approach on a Denial of Service (DoS) attack scenario and present results that demonstrate the benefits of uncertainty-aware decision-making in comparison to using an uncertainty-ignorant approach, both in the presence and absence of uncertainty reduction mechanisms.

1. Introduction

Complex software-intensive systems are increasingly relied on in our society to support tasks in different contexts that are typically characterized by a high degree of *uncertainty*. Self-adaptation [17, 28] is regarded as an effective way to engineer systems that are *resilient* to run time changes despite of the different uncertainties present in their execution environment (e.g., system loads, resource availability, interaction with human actors), goals, or even in the system itself (e.g., the existence and location of faults).

Having inaccurate or missing information could lead a self-adaptive system to make bad decisions that make the system behave worse, rather than improve the system. However, despite the fact that these uncertainties can have a significant negative impact on run-time system behavior [26] (and ultimately, on goal satisfaction), many approaches to engineering self-adaptation do not explicitly represent uncertainty or consider it when deciding what actions to take in the system.