Modelling uncertainty for requirements engineering: The case of surprise

Authors:
Dylan Walton, Nelly Bencomo, Huma Samin

Durham University

Context

A key part of requirements engineering is modelling risk.

It is important to identify where risk might occur and under what conditions.

Difficulties arise when considering many different, interacting variables, and over many timesteps.

A technique that could be used to automatically find and identify risk could greatly speed up this process and prove to be a very helpful to requirement engineers.

Contributions

Surprise classification: Until now surprise has been used to flag for errors in a system, but a comprehensive classification of different surprise amounts has not been done

Surprise comparisons: Two different types of surprise were used, and both were compared for how appropriate they are for surprise classification

Self-adaptive systems (SAS)

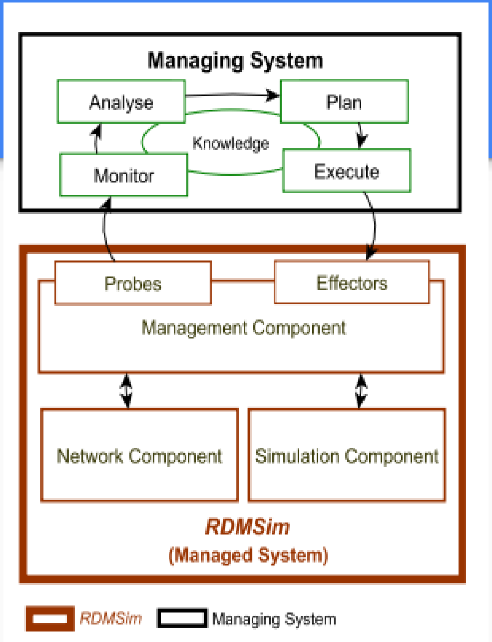
SAS are systems designed to operate under uncertainty and with little to no human interaction.

They need to be capable of dealing with random events arising from actions and changes to the environment.

An exemplar SAS is used to develop the techniques on

Remote data mirror simulator (RDMSim):

- Has 3 non-functional requirements
  - Minimization of cost (MC)
  - Minimization of read/write time (MP)
  - Maximization of reliability (MR)
- Has 7 scenarios



Taken from “RDMSim: An Exemplar for Evaluation and Comparison of Decision-Making Techniques for Self-Adaptation”

Surprise

$$S_{CC} = log_2(\frac{P(m|m_{flat})}{P(m_{flat})})$$

$$S_{BS} = log_2(\frac{P(m|D)}{P(m)})$$

Surprise gives a measure of differences between beliefs, with many different types of surprise designed to attempt capture difference aspects of these differences.

All types of surprise measure the different between some prior (a belief before an observation/event) and a posterior (a belief or after an observation/event)

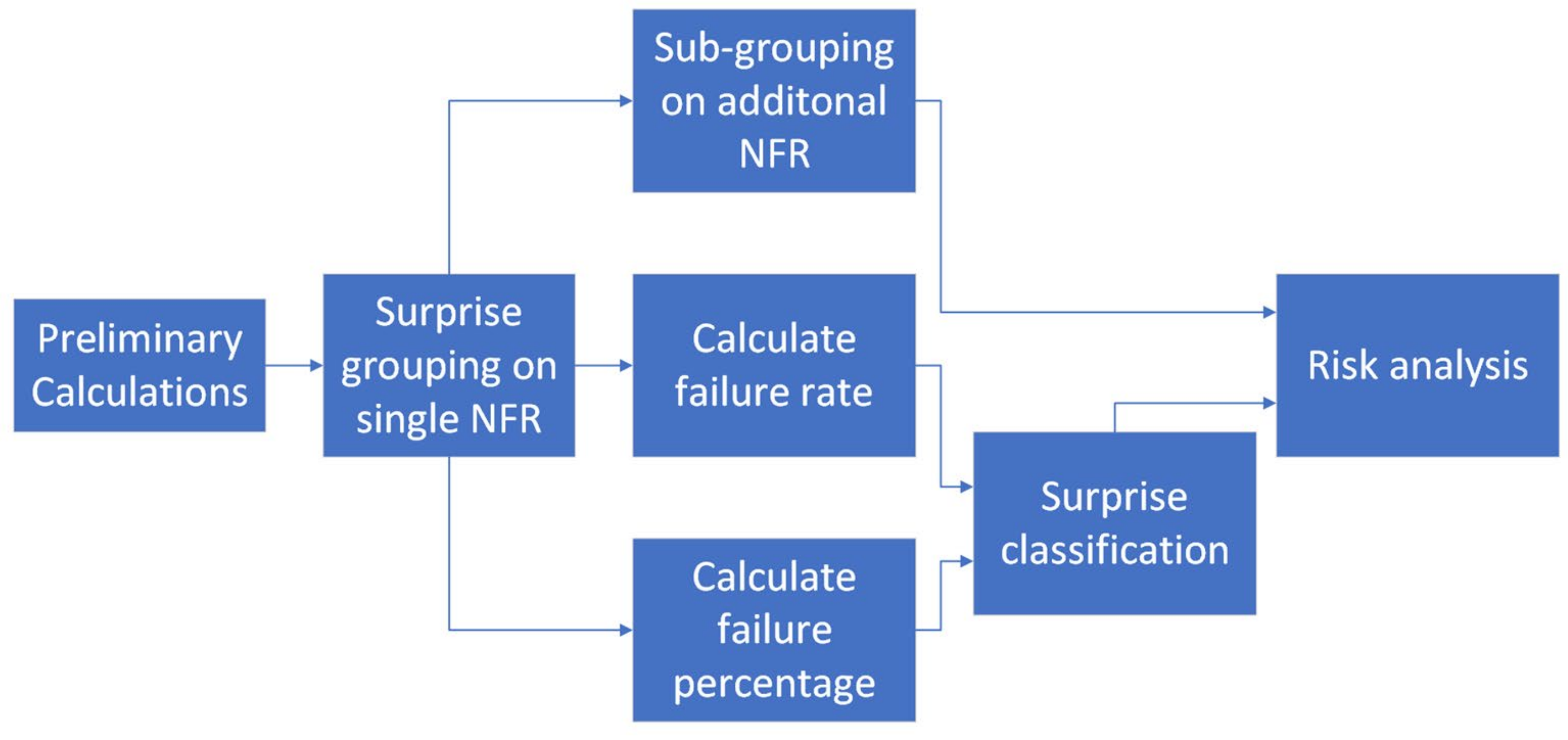
Surprise	Prior	Posterior
Bayesian Surprise	The belief at some random timestep (other than the last)	The belief in the subsequent timestep from the prior
CCS	A flat distribution across the NFR	Any given belief during the runtime

Methodology outline

1. Compute surprise, group surprise based on intervals of 0.1
2. Identify timestamps of first failures, group surprises were failure occurs in intervals of 0.1
3. Compute relevant metrics using techniques (next slide)
4. Perform classification
    a. Initial groupings are done based percentage of failures in a group
    b. Groupings are then altered based on number of interval members and the size of interval compared to all the others

This process is done for both Bayesian and Confidence corrected surprise

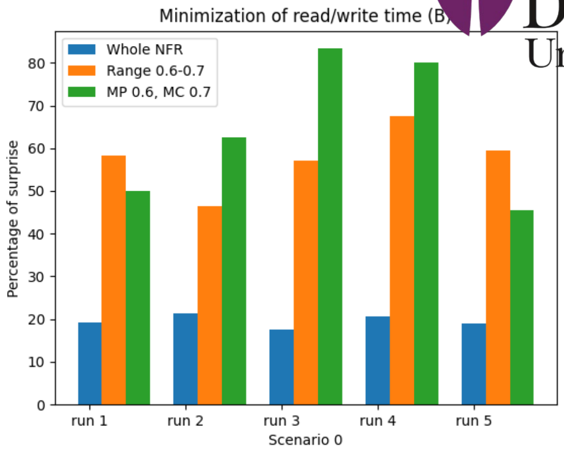
Techniques

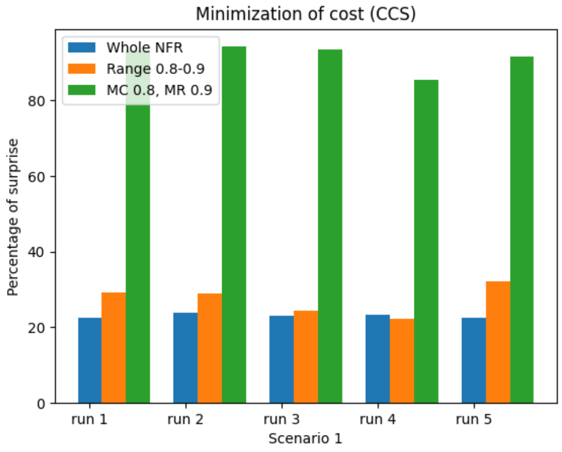

Sub-grouping on additonal NFR
Preliminary Calculations
Surprise grouping on single NFR
Calculate failure rate
Risk analysis
Surprise classification
Calculate failure percentage

Results for Classification

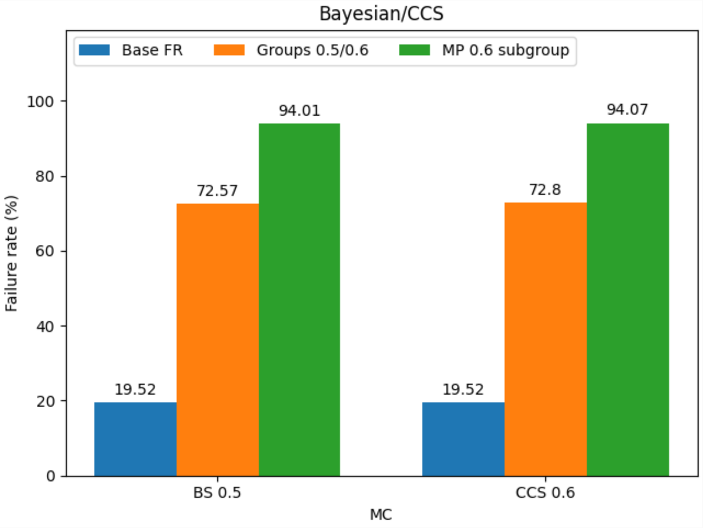
Bayes MC Classifications	
Low	0.3, 0,5
Low-mid	0.7
Middle	0.4
High-mid	-
High	0.6

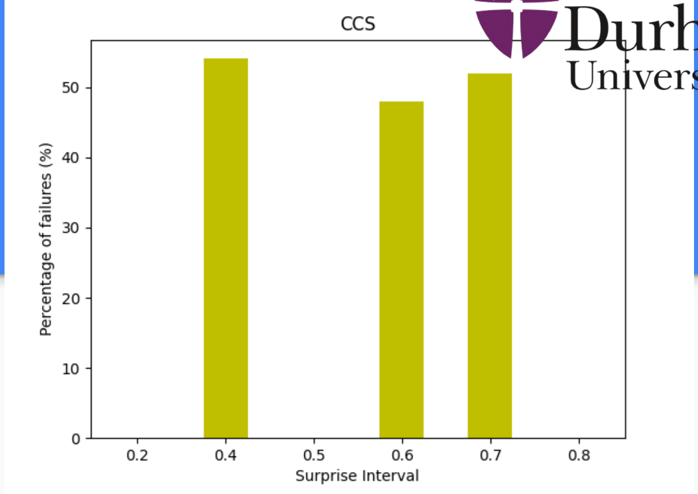
CCS MC Classifications	
Low	0.4
Low-mid	0.7
Middle	0.2
High-mid	-
High	0.5

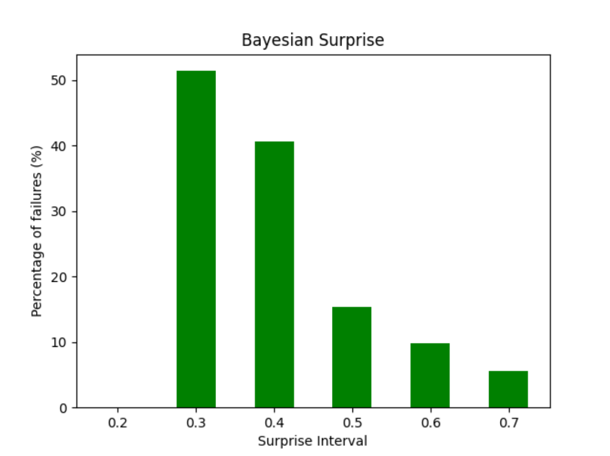




Results for comparison







Discussion and conclusion