

Resource Interaction Failures in Mobile Applications: A Challenge for the SPL Community and Other Perspectives

Euler Marinho

Summary

- An overview of the last submitted paper
- Other perspectives
 - Current research

AN OVERVIEW OF THE LAST SUBMITTED (SHORT) PAPER

Short Paper

- Marinho, E. H., Ferreira, F., Fernandes, E., Diniz, J. P., and Figueiredo, E. “Resource Interaction Failures: A Challenge for SPL Community.” Submitted to the SPLC 2024 Challenge Track.

Introduction

- ❑ Mobile devices have a rich set of resources
- ❑ “Resource” refers to sensors, radios, and user-controlled options
- ❑ User interaction with devices can enable or disable the resources
- ❑ Unexpected application behavior can occur in specific resource settings
- ❑ However, the testing of all input combinations is impracticable

Sampling Strategies

- ❑ Resource interactions are like Feature interactions
- ❑ Resource settings are 14-tuple of resource and state pairs
- ❑ Sampling strategies are alternatives for decreasing the testing effort
- ❑ Random (30), One Enabled (14), One Disabled (14), Most Enabled Disabled (2), Pairwise (8)

Proposed Challenge

- SPLC participants must propose testing strategies for mobile applications
 - Taking resource interactions into account
- The failure detection capability and the effectiveness must be higher than our baseline
 - Increase the number of unique detected failures and minimize the number of tested settings
- Solution efficiency (SE) $SE = \frac{FailingSettings}{TotalSettings}$

Dataset

- 20 Android applications
- 14 target resources
 - Auto Rotate, Battery Saver, Bluetooth, Camera, Do Not Disturb, Location, Mobile Data, Wi-Fi, Accelerometer, Gyroscope, Light, Magnetometer, Orientation, Proximity
- Extended test suites

Evaluation metrics

- Size metrics
 - LOC
 - Number of packages, classes, methods
- Test suite metrics
 - LOC
 - Number of test cases
- Declared resources

Dataset Excerpt

NAME	LOC	#Test Cases	Test LOC	RESOURCES
AnkiDroid	158 K	164	2,770	Cam, MD, Wi-Fi
CovidNow	2 K	21	540	MD, Wi-Fi
Iosched	27 K	9	473	Loc, MD, Wi-Fi
Mixin-Messenger	168 K	160	3,732	BT, Cam, Loc, MD, Wi-Fi
Moonshot	0,455 K	28	464	MD, Wi-Fi
Radio-Droid	22 K	23	1,735	BT, MD, Wi-Fi
WordPress	347 K	115	3,674	Cam, MD, Wi-Fi

BT - Bluetooth
Cam - Camera
Loc - Location
MD - Mobile Data

Test suite instrumentation

- ❑ Functional tests are the target
 - Android APIs for interacting with the device
- ❑ Extension by means of UI Automator
- ❑ Each test class is extended with instrumentation code
- ❑ Before each test case the instrumentation code is executed
- ❑ Test reports are processed

Test Reports

NAME	FAILING SETTINGS	SOLUTION EFFICIENCY	#FAILURES
CovidNow	32	0.47	2
Lockwise	68	1.00	4
Mixin-Messenger	20	0.29	2
Nl-covid19	55	0.81	6
OwnTracks	68	1.00	3
PocketHub	4	0.06	1
SpaceXFollower	68	1.00	4
Threema	33	0.48	1
Vocable	24	0.35	7
WordPress	37	0.54	11

Example of Use

- Settings are provided in CSV files with only enabled resources

**Location, Bluetooth, Battery_Saver,
Do_Not_Disturb, Accelerometer, Light**

Example of Output

- ❑ Vocable is a communication tool for individuals who are speech impaired
- ❑ It uses the ARCore SDK to track the user's head movements
 - To understand where the user is looking on the screen
- ❑ When both Mobile Data and Wi-Fi are disabled (*verifyDefaultTextAppears* test)
 - ARCore fatal exception

CURRENT RESEARCH

Resource Interaction Faults

- Characterize faults behind the failures
 - How to identify faulty classes?
 - Failures are related to the test framework scope
 - Android event-driven nature is a challenge for debugging activities
- Spectrum based Fault Localization
- Extension of our SBES 2023 paper

Resource Interaction Faults

- ❑ Compare faults generated by traditional mutation operators and manually injected faults
- ❑ Faults injected using Bug Fix Patterns (Pan et al. 2009)
 - Using the five most common Bug Fix Patterns (Campos and Maia 2017)

Bug fix patterns (Pan et al. 2009)

□ Change of IF Condition Expression (IF-CC):

```
- if (listBox.getSelectedIndex() == 0)
+ if (listBox.getSelectedIndex() > 0)
```

□ Method Call with different actual parameter values (MC-DAP)

```
- String.getBytes("UTF-8");
+ String.getBytes("ISO-8859-1");
```

Bug fix patterns – Parte 2

- ❑ Method Call with different number of parameters or different types of parameters (MC-DNP)

```
- getSolrQuery(f.getFilter());  
+ getSolrQuery(f.getFilter(), analyzer);
```

- ❑ Change of Assignment Expression (AS-CE)

```
- names[0] = person.getName();  
+ names[0] = employees[0].getName();
```

Bug fix patterns – Parte 3

□ Addition of IF Precondition Check (IF-APC)

```
- repo.getFileContent(path);  
+ if (repo != null && path != null)  
+     repo.getFileContent(path);
```



Questions?