

# Android application stimulation based on human interactions



Adrien Brunelat†, Stefano Zanero‡, Federico Maggi†

†ENS Cachan F-94230 Cachan, France

‡Politecnico di Milano I-20133 Milan, Italy

## Aim

Manually stimulate Android applications to create user interaction traces. Then, combine and use these traces to stimulate **automatically** other similar applications.

## Methods

The PuppetDroid project

PuppetDroid is using a two-timed approach for stimulating applications:

- The **recording** phase:
  - The applications are stimulated by public users.
  - Actions performed are recorded to create an application trace.
- The **rerunning** phase:
  - Applications are triggered using a merged trace from all the records done on the application. If there is no trace for this application, a trace from the most similar application is used.

### XML Elements recognition

The XML recognition allows to track the corresponding interface elements and **correct the stimulation coordinates**. It is also a necessary element of **state recognition** in the process.

PuppetDroid identifies UI elements using the Android SDK's UIAutomator.



Coordinates correction

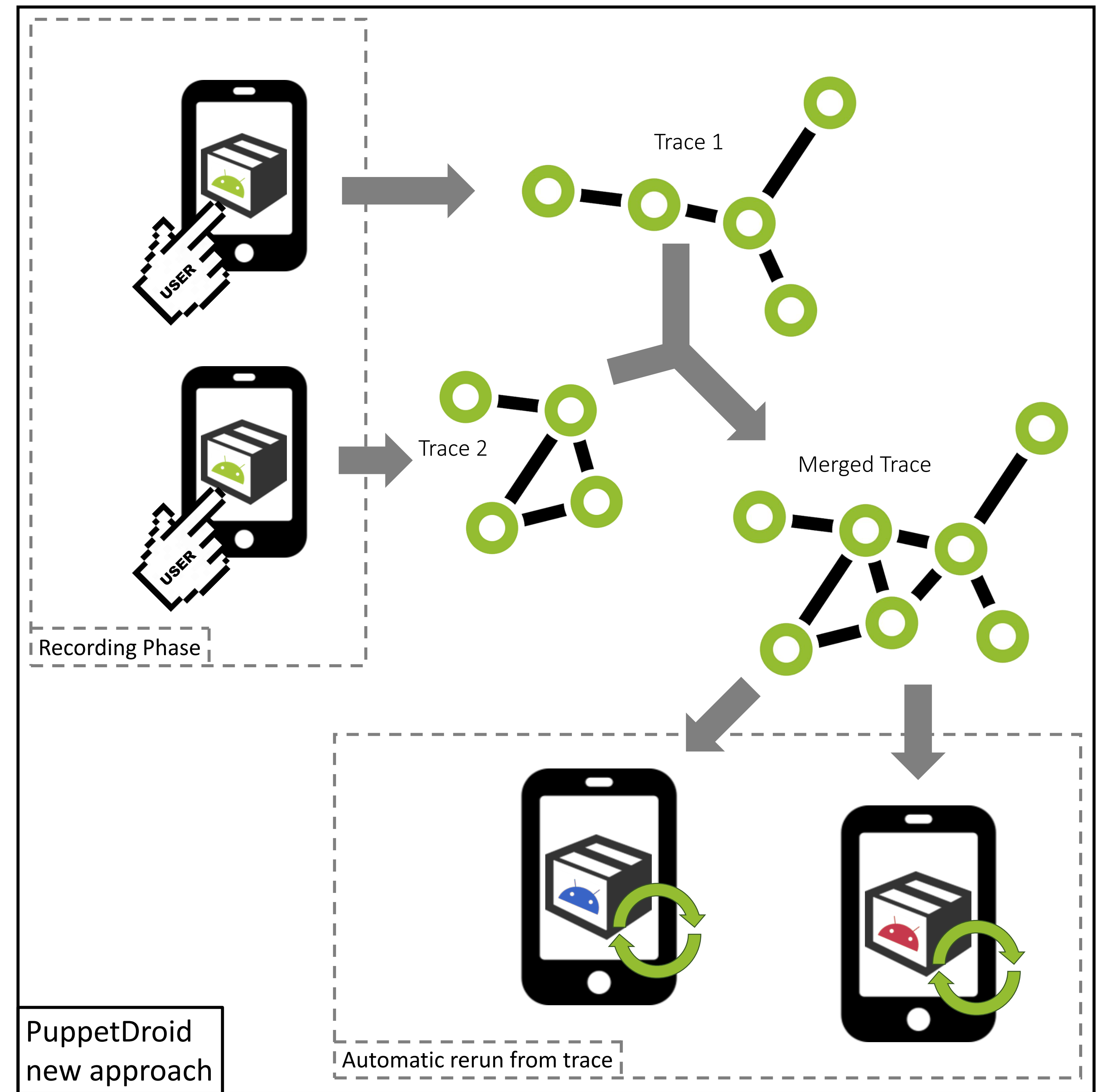
### Graph approach

PuppetDroid is using a **directed state graph** approach for the traces to enable trace merging and smart rerunning.

- Node: Application state
- Edge: User actions to go from a state to another

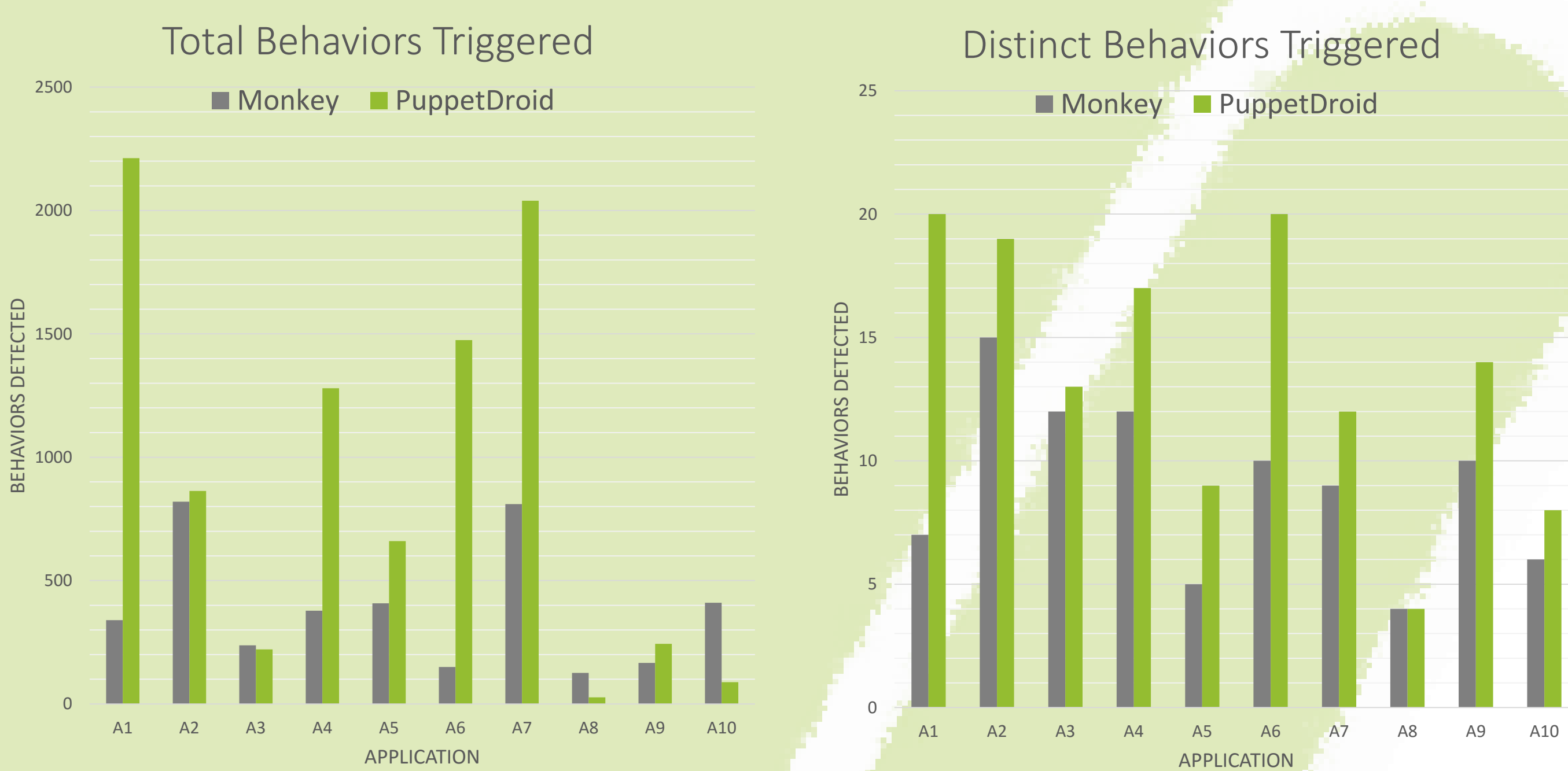
Graphs are then analyzed to find loops and avoid dead-ends during rerunning phase.

The rerunning phase is a navigation through the graph following weighted probabilistic approach that prioritizes the paths that were most stimulated by the users.

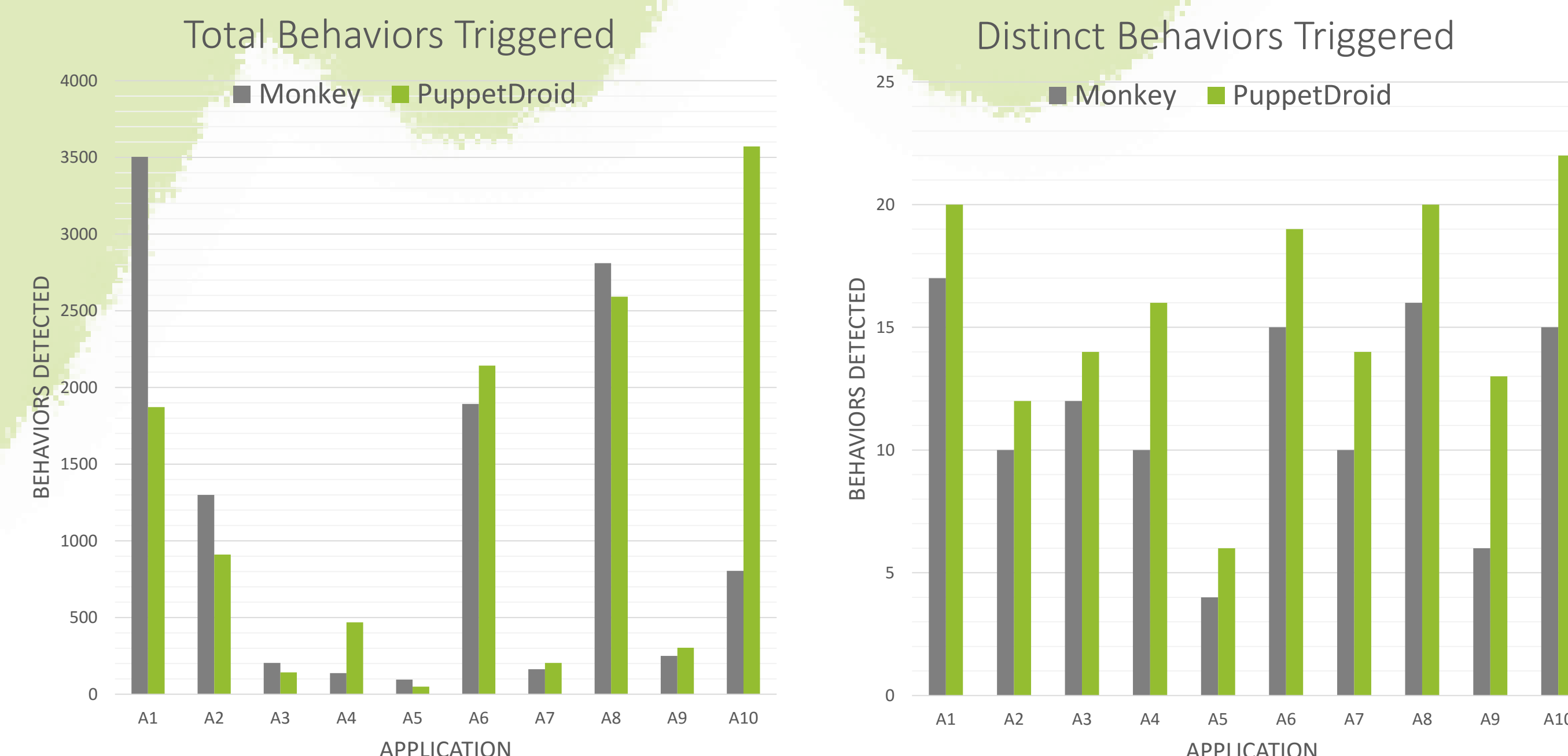


## Results

### Tests on the same application (Small dataset)



### Tests on similar applications (Small dataset)



## Results Interpretation

### Manual small dataset

In some cases, Monkey triggers more total behaviors than PuppetDroid. But, on all the applications, PuppetDroid triggers more distinct behaviors.

With the same application, PuppetDroid is able to trigger 147% times the number of distinct behaviors that Google's Monkey Exerciser do.

When similar applications are chosen manually, the results only go down from 147% to 136% on average with distinct behaviors.

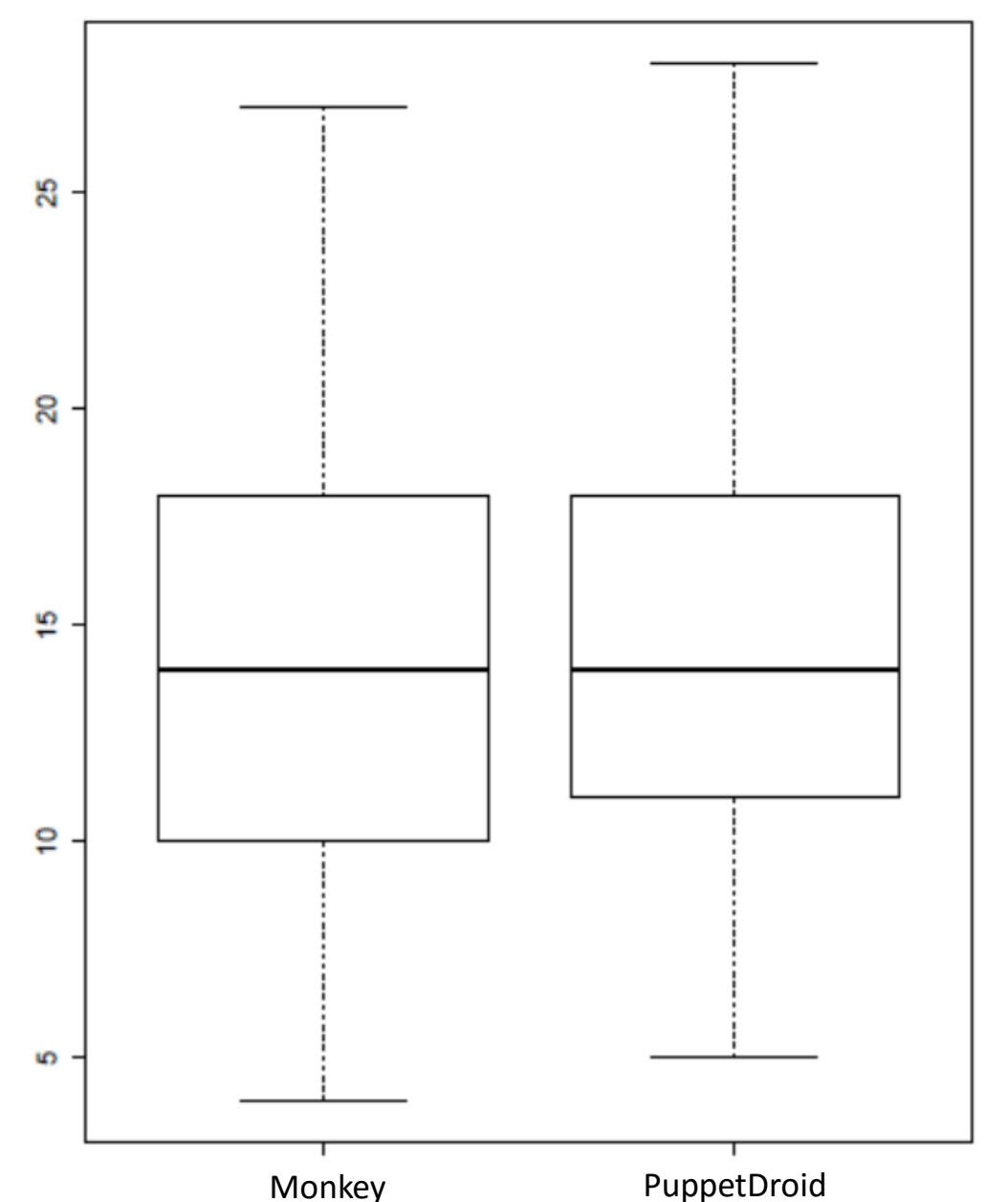
### Automatic large dataset

An algorithm in the project is able to identify similar applications and build a dataset. This one is composed by more than 1000 applications

PuppetDroid performs better than Monkey but not by far.

- It is important to notice that both are not performing best on the same applications.
- Furthermore, the similarity algorithm is less efficient than manual dataset creation. (not possible at this scale)

### Distinct behaviors in large dataset



## Conclusions

When the applications tested are the same or really similar, it appears that PuppetDroid is able to trigger more behaviors than Monkey. It is also able to trigger complicated interface elements thanks to the pre-recorded traces. In addition to that, PuppetDroid reaches a good number of behaviors way faster than Monkey and in a much more reliable way not being fully based on random.

Therefore PuppetDroid appears as a good complementary method to Google's Monkey exerciser.

