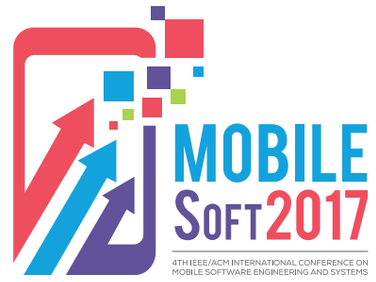


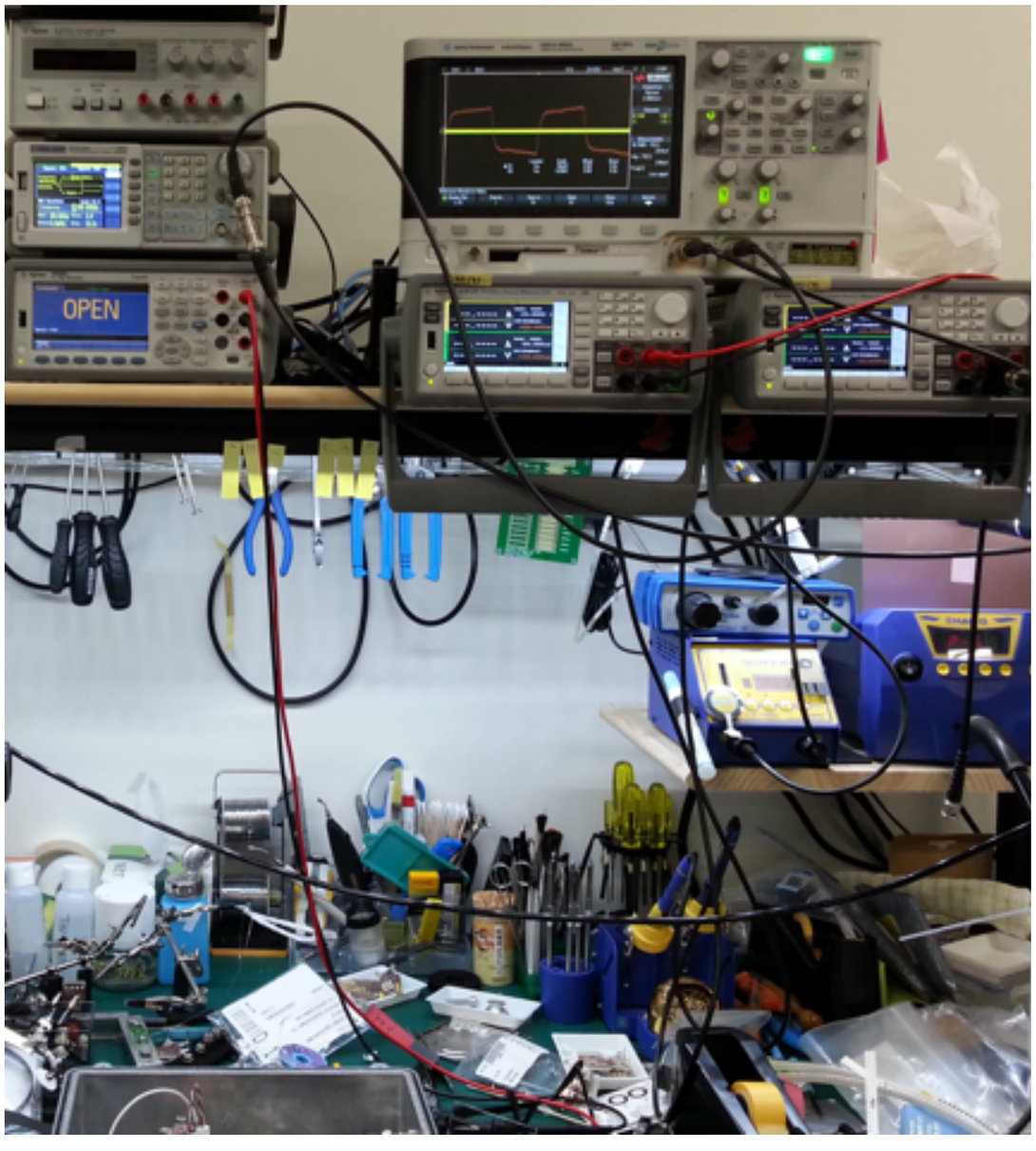
Performance-based Guidelines for Energy-efficient Mobile Applications

Luis Cruz, Rui Abreu

luiscruz@fe.up.pt
rui@computer.org

Motivation

- Mobile and wearable devices are very popular nowadays
- Users expect all-day battery life on their devices
- Impact of energy improvements in mobile applications is hard to measure and time consuming



Hypothesis

Performance based optimizations can be used to ship energy efficient Android applications.

Research Questions

RQ1: Can programming practices be blindly applied in order to improve energy efficiency in an Android application?

RQ2: Do best practices for performance improvement also improve energy efficiency?

RQ3: Do these best practices actually have an impact on real mature Android applications?

Research Questions

RQ1: Can programming practices be blindly applied in order to improve energy efficiency in an Android application?

RQ2: Do best practices for performance improvement also improve energy efficiency?

RQ3: Do these best practices actually have an impact on real mature Android applications?

Research Questions

RQ1: Can programming practices be blindly applied in order to improve energy efficiency in an Android application?

RQ2: Do best practices for performance improvement also improve energy efficiency?

RQ3: Do these best practices actually have an impact on real mature Android applications?

Methodology

A. Android application selection

B. Static analysis and refactoring

C. Generation of automatic UI tests

D. Energy measurement tools setup

E. Experiments execution

F. Data analysis

A. Android application selection

- 6 open-source Apps available at Google Play and/or F-Droid

Loop - Habit Tracker

Writeily Pro

Talalarmo

GnuCash

Simple Gallery

Acrylic Paint

B. Static analysis and refactoring

- Android Lint to detect code smells
- 8 performance-based code smells were studied
- Each code smell was fixed and a new version of the app was generated

Anti-Pattern	Loop - Habit Tracker	Writeily Pro	Talalarmo	GnuCash	Acrylic Paint	Simple Gallery
DrawAllocation	-	-	●	-	●	-
WakeLock	-	-	●	-	-	-
Recycle	-	-	-	●	-	-
ObsoleteLayoutParam	-	-	-	●	-	-
ViewHolder	-	●	-	-	-	-
Overdraw	●	●	-	-	●	●
UnusedResources	●	●	-	-	-	-
UselessParent	-	●	-	-	-	-

C. Generation of automatic UI tests

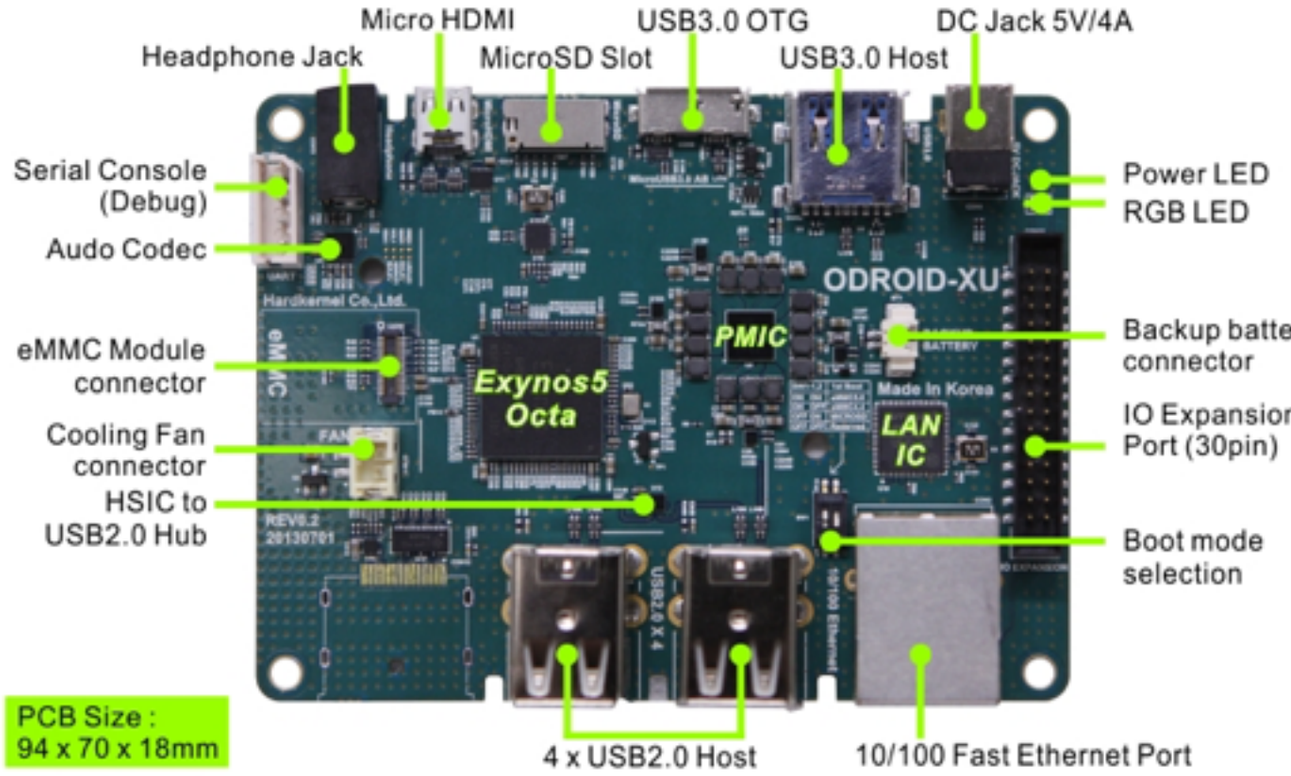
- Scripts to mimic user interaction
- Manually created using *Android View Client*
- Allows replication of experiments

D. Energy measurement tools setup

- Run apps in bare-board computer ODROID

- Power sensors for main CPU, secondary CPU, memory, and GPU

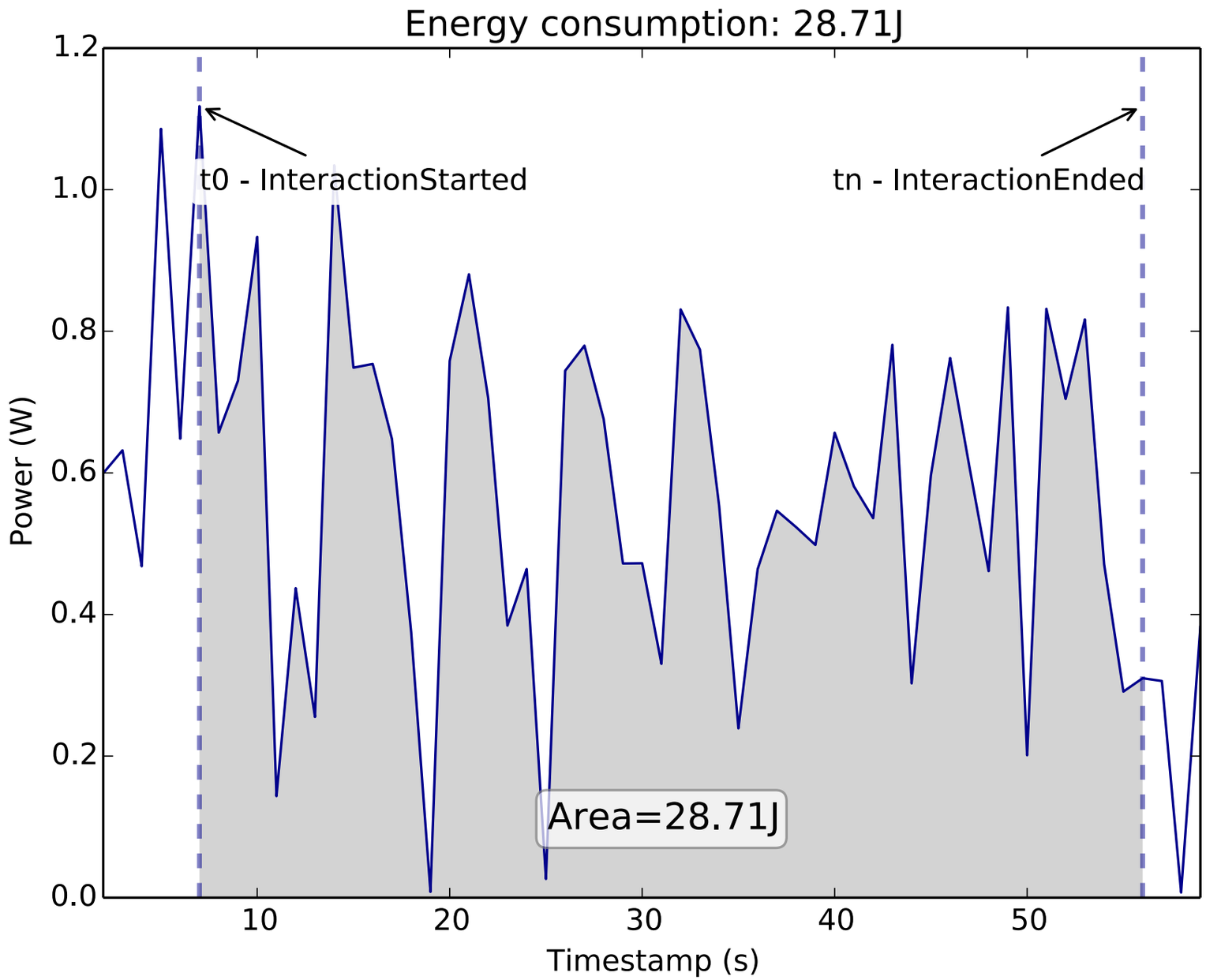
- 4 samples per second



ODROID-XU

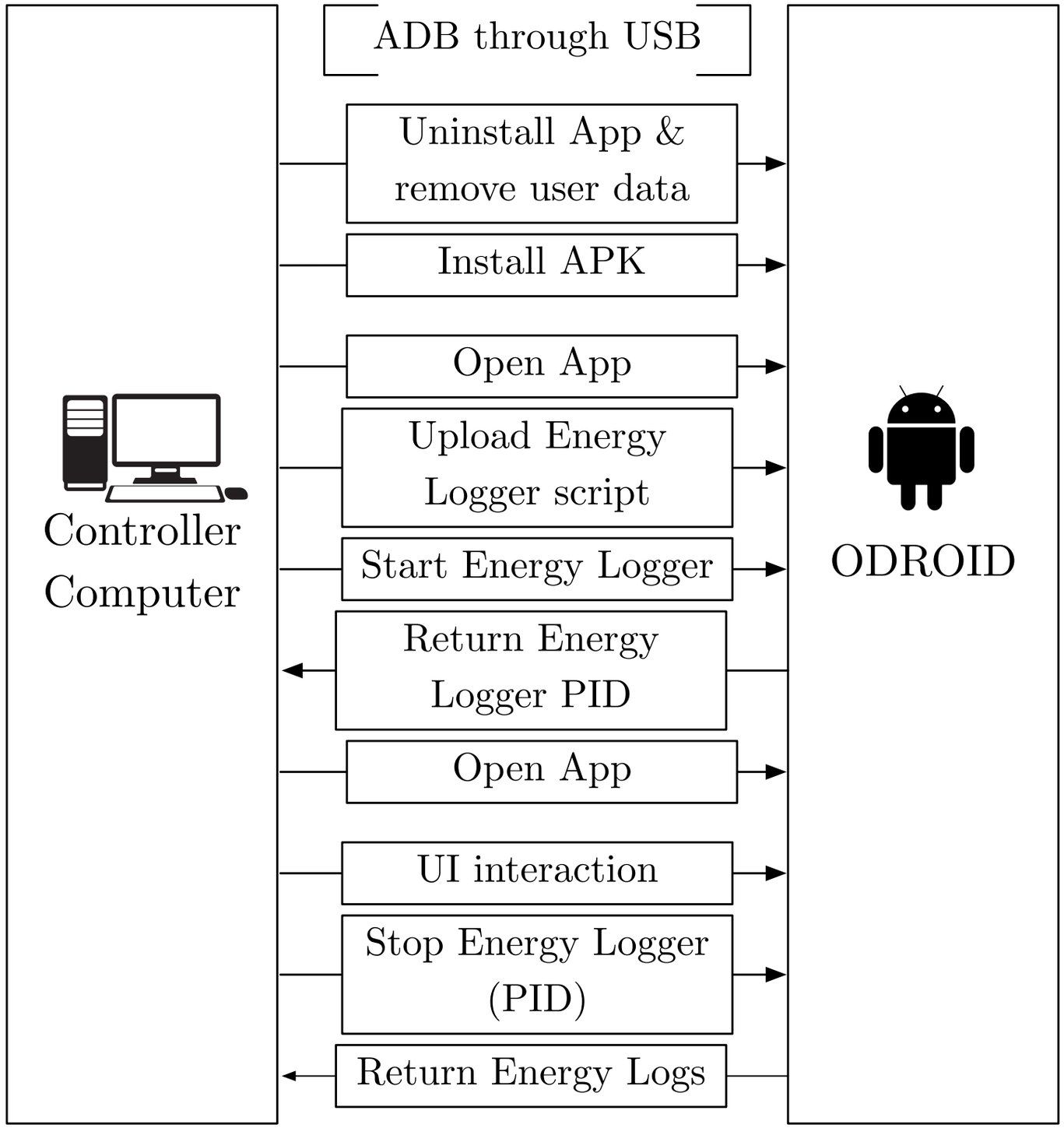
D. Energy measurement tools setup

- Power is an instantaneous measurement (watts)
- Energy is a measurement of power over a period of time (joules)



E. Experiments execution

- For every fixed code smell the experiment was equally executed 30 times for statistical validation



F. Data analysis

- Power readings were down-sampled to 1 second

- Energy consumptions that differ 2 standard deviations from the mean were eliminated

Significance Tests

Welch's t-test results

Application	Pattern	Test	p-value
Loop - Habit Tracker	Overdraw	-0.56	.5784
	UnusedResources	-0.83	.4121
	All	-0.08	.9362
Writeily Pro	Overdraw	-0.10	.9180
	UnusedResources	-0.03	.9790
	ViewHolder	3.02	.0038
	UselessParent	0.20	.8434
	All	2.93	.0049
Talalarmo	DrawAllocation	4.18	.0001
	WakeLock	4.43	$< .0001$
	All	2.16	.0353
GnuCash	ObsoleteLayoutParam	2.57	.0127
	Recycle	2.55	.0140
	All	2.47	.0164
AcrylicPaint	DrawAllocation	0.64	.5221
	Overdraw	45.88	$< .0001$
	All	-5.84	$< .0001$
Simple Gallery	Overdraw	-4.04	.0010

UnusedResource, and **UselessParent** did not provide significant results.

Effect Size

Application	Pattern		MD	Cohen's d	IMP (%)	Savings (min)
Writeily Pro	ViewHolder	↓	-5.39	-0.78	4.50	65
	All	↓	-5.42	-0.76	4.53	65
Talalarmo	DrawAllocation	↓	-0.86	-1.11	1.47	21
	WakeLock	↓	-0.85	-1.17	1.46	21
	All	↓	-0.48	-0.57	0.82	12
GnuCash	ObsoleteLayoutParam	↓	-1.41	-0.67	0.72	10
	Recycle	↓	-1.28	-0.66	0.65	9
	All	↓	-1.53	-0.64	0.78	11
Acrylic Paint	Overdraw	↑	1.42	1.64	-2.26	-33
	All	↑	1.37	1.51	-2.18	-31
Simple Gallery	Overdraw	↑	3.08	1.04	-2.11	-30

View Holder has the biggest impact 😀 while
Overdraw increased energy consumption 😕

Research Questions

RQ1: Can programming practices be blindly applied in order to improve energy efficiency in an Android application?

Yes, apps had energy efficiency improved without changing the feature set and without requiring previous knowledge of the app.

RQ2: Do best practices for performance improvement also improve energy efficiency?

Not necessarily. While five optimizations improved energy efficiency, two did not affect, and one had a negative impact.

RQ3: Do these best practices actually have an impact on real mature Android applications?

Yes, three out of six real apps improved energy efficiency.

Conclusions & Future Work

- Anti-patterns ***ViewHolder***, ***DrawAllocation***, ***WakeLock***, ***ObsoleteLayoutParam***, and ***Recycle*** have to be considered when developing energy-efficient apps

- Extend the study to other optimizations

- Automatic refactoring (Autorefactor, FB pfff, Walkmod, Kadabra?)

- Label mobile applications with respect to energy efficiency

Hypothesis

Performance based optimizations can be used to ship energy efficient Android applications.

Effect Size

Application	Pattern		MD	Cohen's d	IMP (%)	Savings (min)
Writeily Pro	ViewHolder	↓	-5.39	-0.78	4.50	65
	All	↓	-5.42	-0.76	4.53	65
Talalarmo	DrawAllocation	↓	-0.86	-1.11	1.47	21
	WakeLock	↓	-0.85	-1.17	1.46	21
	All	↓	-0.48	-0.57	0.82	12
GnuCash	ObsoleteLayoutParam	↓	-1.41	-0.67	0.72	10
	Recycle	↓	-1.28	-0.66	0.65	9
	All	↓	-1.53	-0.64	0.78	11
Acrylic Paint	Overdraw	↑	1.42	1.64	-2.26	-33
	All	↑	1.37	1.51	-2.18	-31
Simple Gallery	Overdraw	↑	3.08	1.04	-2.11	-30

View Holder has the biggest impact 😀 while **Overdraw** increased energy consumption 😕

Research Questions

RQ1: Can programming practices be blindly applied in order to improve energy efficiency in an Android application?

Yes, apps had energy efficiency improved without changing the feature set and without requiring previous knowledge of the app.

RQ2: Do best practices for performance improvement also improve energy efficiency?

Not necessarily. While five optimizations improved energy efficiency, two did not affect, and one had a negative impact.

RQ3: Do these best practices actually have an impact on real mature Android applications?

Yes, three out of six real apps improved energy efficiency.

Performance-based Guidelines for Energy-efficient Mobile Applications

Luis Cruz, Rui Abreu

Full Paper

All source code and collected data