



Titre: Optimizing user experience in choosing android applications
Title:

Auteurs: Rubén Saborido, Giovanni Beltrame, Foutse Khomh, Enrique Alba, & Giuliano Antoniol
Authors:

Date: 2016

Type: Rapport / Report

Référence: Saborido, R., Beltrame, G., Khomh, F., Alba, E., & Antoniol, G. (2016). Optimizing user experience in choosing android applications. (Rapport technique n° EPM-RT-2016-03). <https://publications.polymtl.ca/2971/>
Citation:

Document en libre accès dans PolyPublie

Open Access document in PolyPublie

URL de PolyPublie: <https://publications.polymtl.ca/2971/>
PolyPublie URL:

Version: Version officielle de l'éditeur / Published version

Conditions d'utilisation: Tous droits réservés / All rights reserved
Terms of Use:

Document publié chez l'éditeur officiel

Document issued by the official publisher

Institution: École Polytechnique de Montréal

Numéro de rapport: EPM-RT-2016-03
Report number:

URL officiel:
Official URL:

Mention légale:
Legal notice:

EPM-RT-2016-01

**OPTIMIZING USER EXPERIENCE IN
CHOOSING ANDROID APPLICATIONS**

Rubén Saborido¹, Giovanni Beltrame¹, Foutse Khomh¹,
Enrique Alba², Giuliano Antoniol¹
Département de génie informatique et génie logiciel¹
École Polytechnique de Montréal
Department of Computer Science²
University of Malaga

March 2016

Poly

EPM-RT-2016-03

Optimizing User Experience in Choosing Android Applications

Rubén Saborido¹, Giovanni Beltrame¹, Foutse Khomh¹,
Enrique Alba², Giuliano Antoniol¹
Département de génie informatique et génie logiciel¹
École Polytechnique de Montréal
Department of Computer Science²
University of Malaga

March 2016

©2016

Rubén Saborido, Giovanni Beltrame, Foutse Khomh,
Enrique Alba, Giuliano Antoniol
Tous droits réservés

Dépôt légal :

Bibliothèque nationale du Québec, 2016
Bibliothèque nationale du Canada, 2016

EPM-RT-2016-03

Optimizing User Experience in Choosing Android Applications

par : Rubén Saborido¹, Giovanni Beltrame¹, Foutse Khomh¹,
Enrique Alba², Giuliano Antoniol¹

Département de génie informatique et génie logiciel¹

École Polytechnique de Montréal

Department of Computer Science²

University of Malaga

Toute reproduction de ce document à des fins d'étude personnelle ou de recherche est autorisée à la condition que la citation ci-dessus y soit mentionnée.

Tout autre usage doit faire l'objet d'une autorisation écrite des auteurs. Les demandes peuvent être adressées directement aux auteurs (consulter le bottin sur le site <http://www.polymtl.ca/>) ou par l'entremise de la Bibliothèque :

École Polytechnique de Montréal
Bibliothèque – Service du Prêt entre bibliothèques
Case postale 6079, Succursale «Centre-Ville»
Montréal (Québec)
Canada H3C 3A7

Téléphone :

(514) 340-4846

Télécopie :

(514) 340-4026

Courrier électronique :

biblio.sfd@courriel.polymtl.ca

Ce rapport technique peut-être repéré par auteur et par titre dans le catalogue de la Bibliothèque :
<http://vsmart.biblio.polymtl.ca/epmvw/Vubis.csp?Profile=DEFAULT&OpacLanguage=fr>

Optimizing User Experience in Choosing Android Applications

Technical Report

Paper presented in SANER 2016
23rd IEEE International Conference on
Software Analysis, Evolution, and Reengineering.

Rubén Saborido, Giovanni Beltrame, Foutse Khomh,
Enrique Alba, Giuliano Antoniol

Abstract

Why is my cell phone battery already low? How did I use almost all the data of my monthly Internet plan? Is my recently released new application more efficient than similar competing applications? These are not easy questions to answer. Different applications implementing similar or identical functionalities may have different energy consumptions.

In the paper associated to this technical report we present a recommendation system aimed at helping users and developers alike. We help users to choose optimal sets of applications belonging to different categories (*eg.* browsers, e-mails, cameras) while minimizing energy consumption, transmitted data, and maximizing application rating. We also help developers by showing the relative placement of their application's efficiency with respect to selected others. When the optimal set of applications is computed, it is leveraged to position a given application with respect to the optimal, median and worst application in its category (*eg.* browsers).

Out of eight categories we selected 144 applications, manually defined typical execution scenarios, collected the relevant data, and computed the Pareto optimal front solving a multi-objective optimization problem. We report evidence that, on the one hand, ratings do not correlate with energy efficiency and data frugality. On the other hand, we show that it is possible to help developers understanding how far is a new Android application power consumption and network usage with respect to optimal applications in the same category.

From the user perspective, we show that choosing optimal sets of applications, power consumption and network usage can be reduced by 16.61% and 40.17%, respectively, in comparison to choosing the set of applications that maximizes only the rating.

This document is the technical report associated to the paper “Optimizing User Experience in Choosing Android Applications” [3]. Here we extent the original paper answering some questions related to the optimization process and giving all the figures and statistical tests generated in our experiments. Therefore, this document can be considered as an appendix of the original paper.

1 Optimization problem solved by ADAGO

Here we summarize the multi-objective optimization problem described in the paper and solved by ADAGO (our approach to found optimal sets of Android applications minimizing power consumption and network usage, and maximizing the applications rating). Let $\mathcal{C} = \{C_1, \dots, C_N\}$ be a set of categories. Further assume that, for each category C_i , a set A_i of applications has been selected. In other words, $\mathcal{A} = \{A_1, \dots, A_N\}$ is the application set of sets. An element \mathbf{x} of the search space F , $\mathbf{x} = (x_1, \dots, x_N)$, is a set of applications where x_j is an application selected from A_j and an application from each category in \mathcal{C} has been chosen. In other words, a solution must contain an application per category and all categories must be present.

ADAGO goal is to analyze the trade-off between power consumption, network usage and global rating, and it models the following combinatorial multi-objective optimization problem:

$$\begin{aligned} &\text{optimize} \quad [power(\mathbf{x}), network(\mathbf{x}), rating(\mathbf{x})] \\ &\text{s.t.} \quad \mathbf{x} \in F \end{aligned} \tag{1}$$

Given a solution \mathbf{x} , the objective functions used in (1) are calculated as follow:

$$power(\mathbf{x}) = \frac{\sum_{i=1}^N power(x_i)}{N} \tag{2}$$

$$network(\mathbf{x}) = \frac{\sum_{i=1}^N network(x_i)}{N} \tag{3}$$

$$rating(\mathbf{x}) = \frac{\sum_{i=1}^N rating(x_i)}{N} \tag{4}$$

In equations (5) and (6), $power(x_i)$ and $network(x_i)$ are the average values of power (in Watts) and network usage (in megabytes) for application x_i in a certain number of runs and for a given number of exercised application functionalities.

2 Choosing an optimal app by category is an optimal solution?

It is normal to ask if an optimal application is chosen by each category the final set of applications is an optimal solution. Here we show that choosing an optimal application per category does not warranty an optimal solution for our problem.

Let us suppose we have two categories, *Category A* and *Category B*, and two apps in each of them. The objective values associated to these apps are showed in Table 1.

Table 1: Pareto optimal apps per category

	Category A			Category B		
Objective	Power	Network	Rating	Power	Network	Rating
App1	1	4	4	5	2	4
App2	4	1	4	1	3	4

Given that we have two solutions per category and two categories, there exist $2^2 = 4$ possible combinations of applications. For each combination the objective values are calculated considering the following equations (or using the average dividing by N like in the paper, but we use these ones for simplicity):

$$power(\mathbf{x}) = \sum_{i=1}^C power(x_i) \quad (5)$$

$$network(\mathbf{x}) = \sum_{i=1}^C network(x_i) \quad (6)$$

$$rating(\mathbf{x}) = \sum_{i=1}^C rating(x_i) \quad (7)$$

All the combinations and its respective objective values are showed in Table 2. As is showed, the solution associated to *Combination 1* is not Pareto optimal because it is dominated by the solution associated to *Combination 4*. So, we conclude that *Combination 1* is not a Pareto optimal solution and it is not consider a solution for us.

Table 2: All the possible combinations

	Category A			Category B			Solution		
Objective	Power	Network	Rating	Power	Network	Rating	Power	Network	Rating
Combination 1	1	4	4	5	2	4	6	6	8
Combination 2	1	4	4	1	3	4	2	7	8
Combination 3	4	1	4	5	2	4	9	3	8
Combination 4	4	1	4	1	3	4	5	4	8

3 Solving the Problem using Metaheuristics

If the number of categories or the number of applications per category is greater the search space could be too large to be explored exhaustively and metaheuristics would be needed. In these cases, the multi-objective optimization problem (1) can be solved using EMO algorithms, as *NSGA-II*. To check this fact, problem (1) is defined and included in *jMetal*[2], an object-oriented Java-based framework for multi-objective optimization with meta-heuristics. Considering the parameters settings described in Table 3, *NSGA-II* is run five times for each value of the crossover probability, what supposes five independent runs for each P_x value. These well known parameters and operators are commented in [1]. Finally, Pareto optimal solutions generated in each run are combined and filtered applying the Pareto dominance relation to generate a global reference front, which contains the set of best solutions considering all the runs.

Table 3: Parameters settings for *NSGA-II*

Parameter	Value
<i>Population size</i>	200
<i>Generations</i>	300
<i>Crossover operator</i>	Single point crossover
<i>Crossover probability (P_x)</i>	0.3, 0.5, 0.7, 0.9
<i>Mutation operator</i>	Flip mutation
<i>Mutation probability (P_m)</i>	$1/C = 0.125$
<i>Selection operator</i>	Binary tournament

The *Single Point* crossover operator is used because it is one of the simplest crossover operators and it works reasonably fine in combinatorial problems. When two parents are selected, with a probability of P_x the operator is used to create new individuals. It selects a point on both parents and all data beyond that point in either individual is swapped between the two parents. The resulting solutions or individuals are the offspring. Considering the mutation, the *Flip* mutation operator is used. It changes the value of a gene in the individual, with a probability of P_m , with a new value generated randomly in the lower and upper bounds range. The *Binary tournament* selection operator is used to select individuals in the population to create the offspring. This operator selects two solutions randomly in the population and chooses the best one, of one of them with a probability of 0.5 if they are equivalents. The final ADAGO Pareto optimal front obtained by *NSGA-II* is showed in Figure 1 where it is compared to the Pareto optimal front generated exhaustively after the search space reduction.

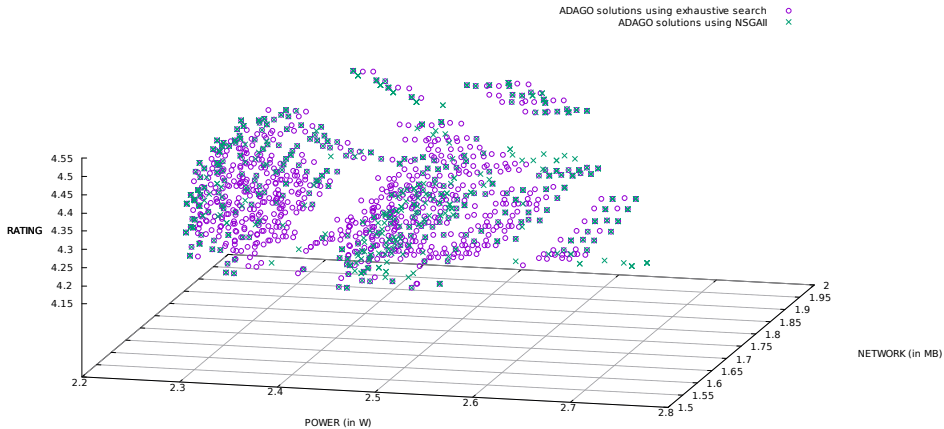


Figure 1: Comparison of ADAGO solutions generated by exhaustive search VS *NSGA-II*.

NSGA-II is able to generate a subset of Pareto optimal solutions for problem (1) and, therefore, we can consider this kind of algorithms when exhaustive search is not valid.

4 Statistical Tests

Table 4 reports, for all the categories, the results of *Wilcoxon signed-rank* and the *Cliff-Delta* effect size. Bold values for p -value indicate that the comparison is statistically significant (at 95%) after *Holmes* correction. For each category (first table column), the maximum rating is reported which is the rating associated to the application chosen by the user. One can notice (third column) that the Pareto front contains applications with the user selected application. Actually (see the *Music Players* category) the user selection is sometime an optimal application. This happened for three categories. In such cases the p -value is not significant and the effect size is zero. What we observe in Table 4 is that not always the ADAGO solution has the lower energy (data overhead) consumption. Anyway, as we expected, either the user selection is optimal or ADAGO applications have either lower energy footprint or lower data footprint with very significant p -values. Furthermore, the effect size is almost always very high. Cliff delta is considered to be negligible for values below 0.147, small below 0.33, medium below 0.475 and large otherwise. We thus conclude that when (if remove the special case when the user selection is optimal) energy (data overhead) of ADAGO’s application is lower than the user selected, this is statistically significant and with a large effect size.

Table 4: Statistical test comparing ADAGO and the app selected by the user in each category.

Category (max. Rating)	Application	Rating	Power		Network	
			p-val	clifd	p-val	clifd
<i>Browsers</i> (4.60)	com.apusapps.browser	4.60	0.8299	0.2050	0.0000	0.9500
	com.ksmobile.cb	4.60	0.9042	0.0250	0.0000	0.9500
	com.opera.mini.native	4.40	0.0448	0.4650	0.0000	0.9500
	com.UCMobile.intl	4.50	0.8299	0.1750	0.0000	0.9500
<i>Cameras</i> (4.50)	cn.jingling.motu.photowonder	4.30	0.0000	-1.0000	0.0000	0.9444
	com.arcsoft.perfect365	4.20	1.0000	-0.1852	0.0000	0.9444
	com.finnal.photoeditor	4.20	0.0000	-0.9938	0.0000	0.9444
	com.kvadgroup.photostudio	4.10	0.0000	-0.9444	0.0000	0.9444
	com.lyrebirdstudio.mirror	4.20	1.0000	0.1111	0.0000	0.9444
	com.picsart.studio	4.40	0.0002	0.7500	1.0000	-0.0185
	com.pixlr.express	4.40	1.0000	0.0679	0.0000	0.9444
	com.roidapp.photogrid (*)	4.50	1.0000	0.0000	1.0000	0.0000
	com.seventeenmiles.sketch	4.30	0.0000	-0.9877	0.0000	0.9444
	com.studio8apps.instasizenocrop	4.40	0.0000	-0.9877	0.0000	0.9444
<i>Emails</i> (4.30)	com.fsck.k9 (*)	4.30	1.0000	0.0000	NaN	0.0000
	com.google.android.gm	4.30	0.0159	0.8000	0.0040	-1.0000
<i>Flash Lights</i> (4.70)	com.apusapps.tools.flashtorch	4.30	0.0006	0.6850	0.0000	0.9500
	com.devuni.flashlight	4.40	0.0413	0.4700	0.0000	0.9500
	com.intellectualflame.ledflashlight.washer	4.50	0.0013	0.6500	0.0085	0.5300
	com.rayg.flashlightfree	3.80	0.0000	0.8700	0.0000	0.9500
	com.rvappstudios.flashlight	4.30	0.0000	-0.9500	0.0000	0.9500
	com.teslacoilsw.flashlight	4.50	0.4251	0.2550	0.0000	0.9500
	com.zeroneapps.flashlight	3.90	0.0000	0.7950	0.0000	0.9125
	flashlight.led.clock	4.40	0.4251	0.2750	0.0000	0.9500
	goldenshoretechnologies.brightestflashlight.free (*)	4.70	1.0000	0.0000	1.0000	0.0000
<i>Music Players</i> (4.60)	cn.voilet.musicplaypro	4.40	0.0000	0.9500	0.0000	-1.0000
	com.aimp.player	4.50	0.0000	0.8550	0.0000	0.9500
	com.n7mobile.nplayer	4.50	0.0064	0.5350	0.0000	0.9500
	com.tbigh.playerprotrial (*)	4.60	1.0000	0.0000	1.0000	0.0000
<i>News</i> (4.50)	com.mobilesrepublic.appy	4.50	0.0032	0.5350	0.0000	0.9500
	net.aljazeera.english	4.10	0.0019	0.5950	0.0000	0.9500
<i>Video Players</i> (4.50)	com.kmplayer	4.20	0.0037	0.5950	0.0001	-0.7850
	org.videolan.vlc	4.40	0.0586	0.4300	0.0008	0.6225
	video.player.audio.player.music (*)	4.50	1.0000	0.0000	1.0000	0.0000
	videoplayer.mediaoplayer.hdplayer	4.50	0.8899	0.1450	1.0000	-0.0900
<i>Weather</i> (4.50)	com.droid27.senseflipclockweather	4.20	0.0154	0.5100	0.0000	0.9175
	com.droid27.transparentclockweather	4.30	0.0154	0.5100	0.0000	0.8875
	com.gau.go.launcherex.gowidget.weatherwidget	4.50	0.1572	0.2650	0.0000	0.9500
	com.weather.Weather	4.30	0.0000	0.7550	0.0071	-0.5000
	de.wetteronline.wetterapp	4.50	0.0057	0.5750	0.0000	0.8200
	local.weather.forecast.pro	4.10	0.0052	0.5900	0.0000	0.9500

5 Additional Figures and Charts

Figures 2-7 show the average power consumption and the total network usage, in 20 runs, for each application in the different categories.

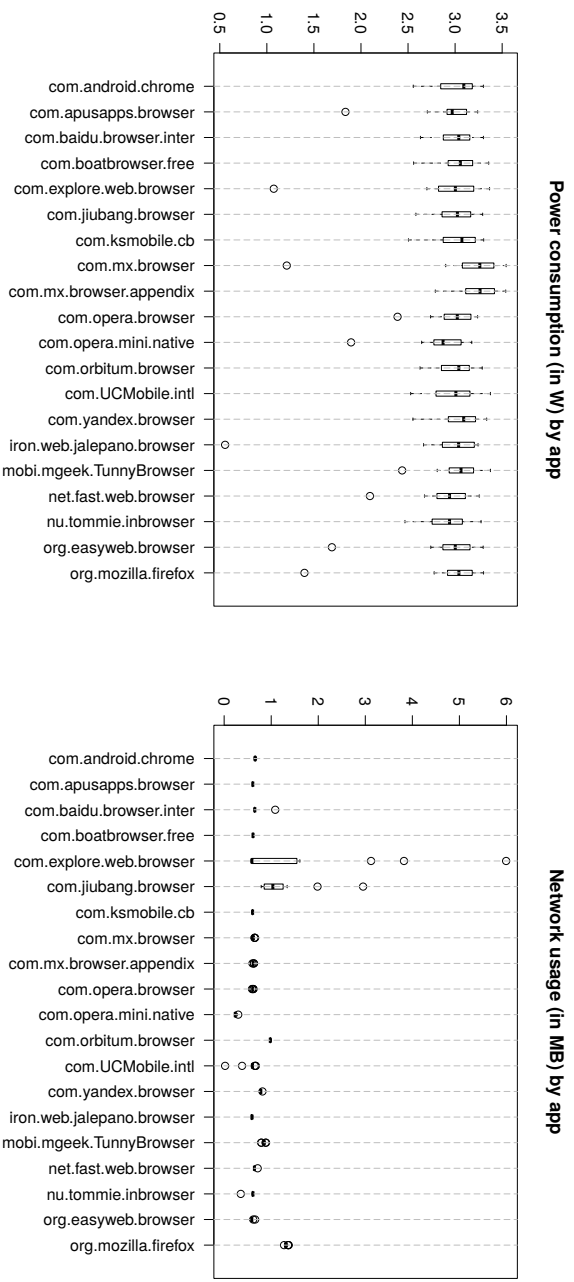


Figure 2: Average power consumption and network usage for apps in the *Browsers* category.

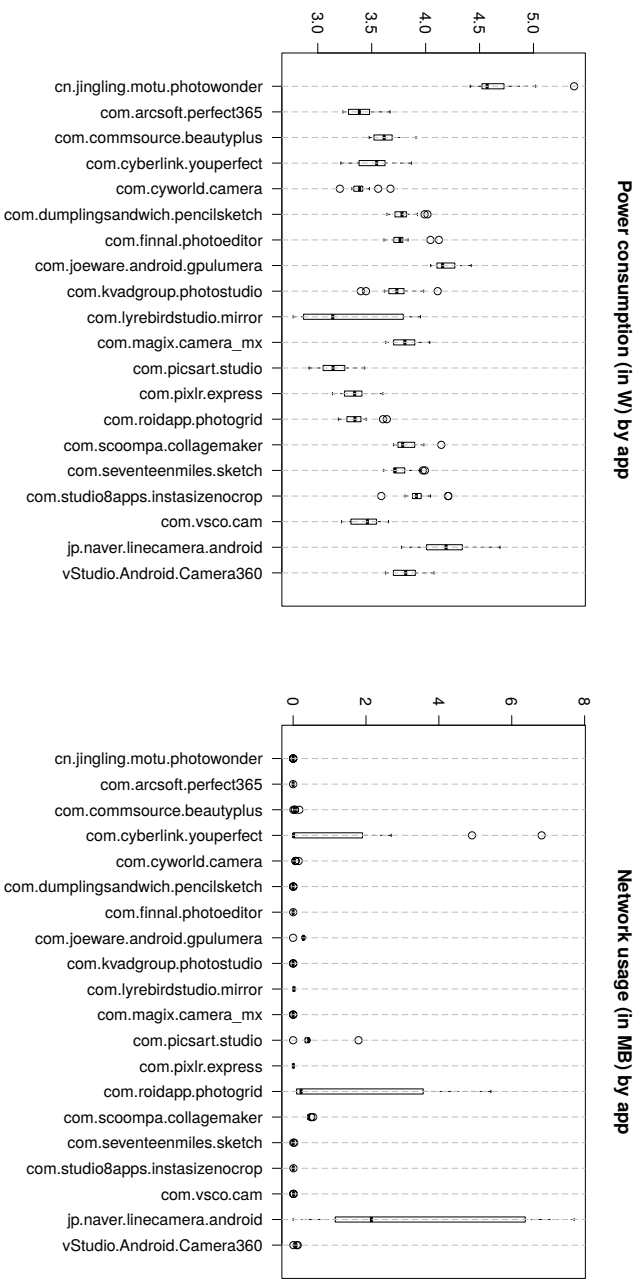


Figure 3: Average power consumption and network usage for apps in the *Cameras* category.

To get a global vision per category, Figure 10 shows the average power consumption and the average network usage for selected applications in each category.

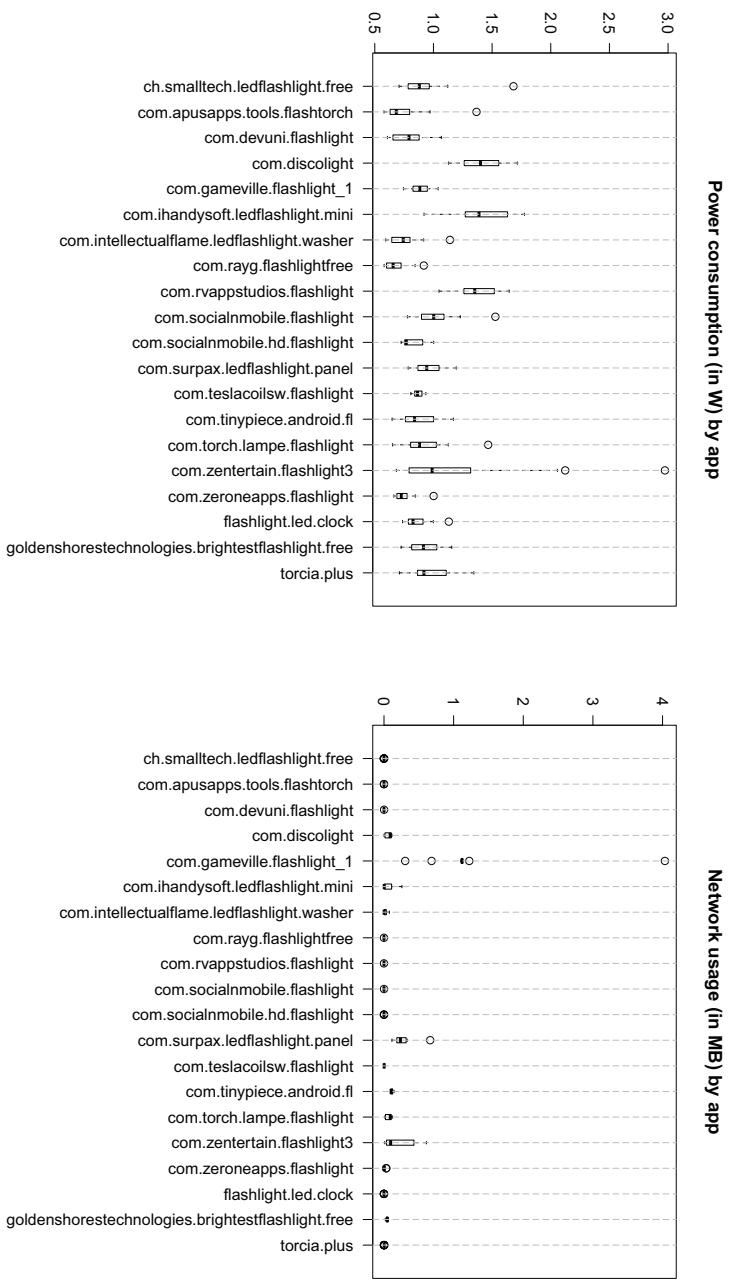


Figure 4: Average power consumption and network usage for apps in the *Flashlights* category.

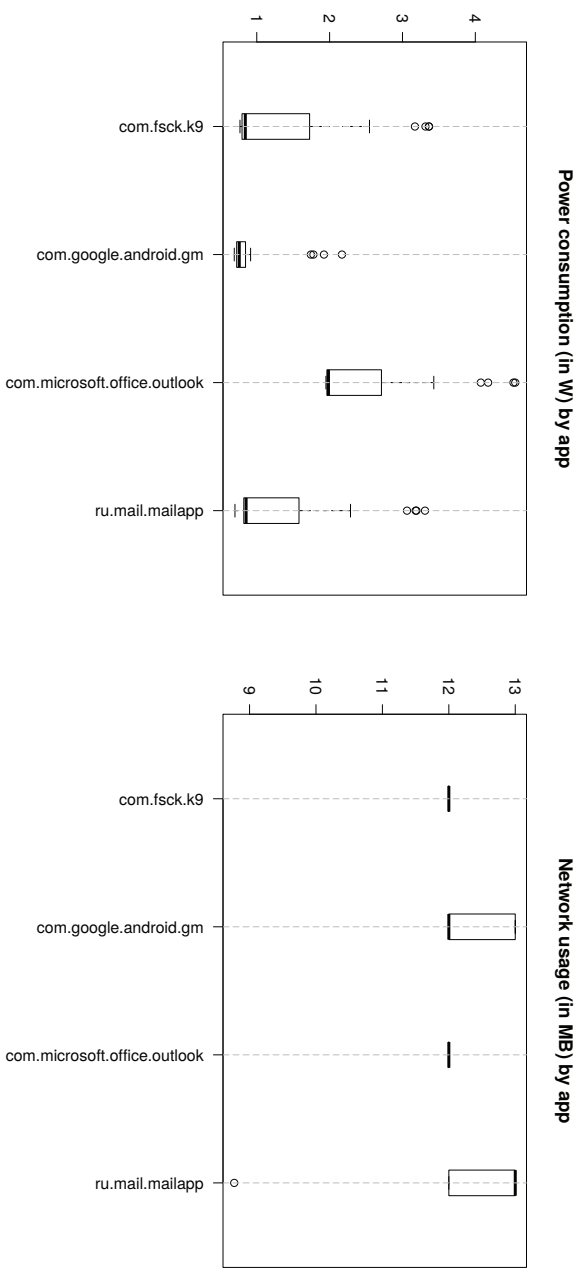


Figure 5: Average power consumption and network usage for apps in the *Emails* category.

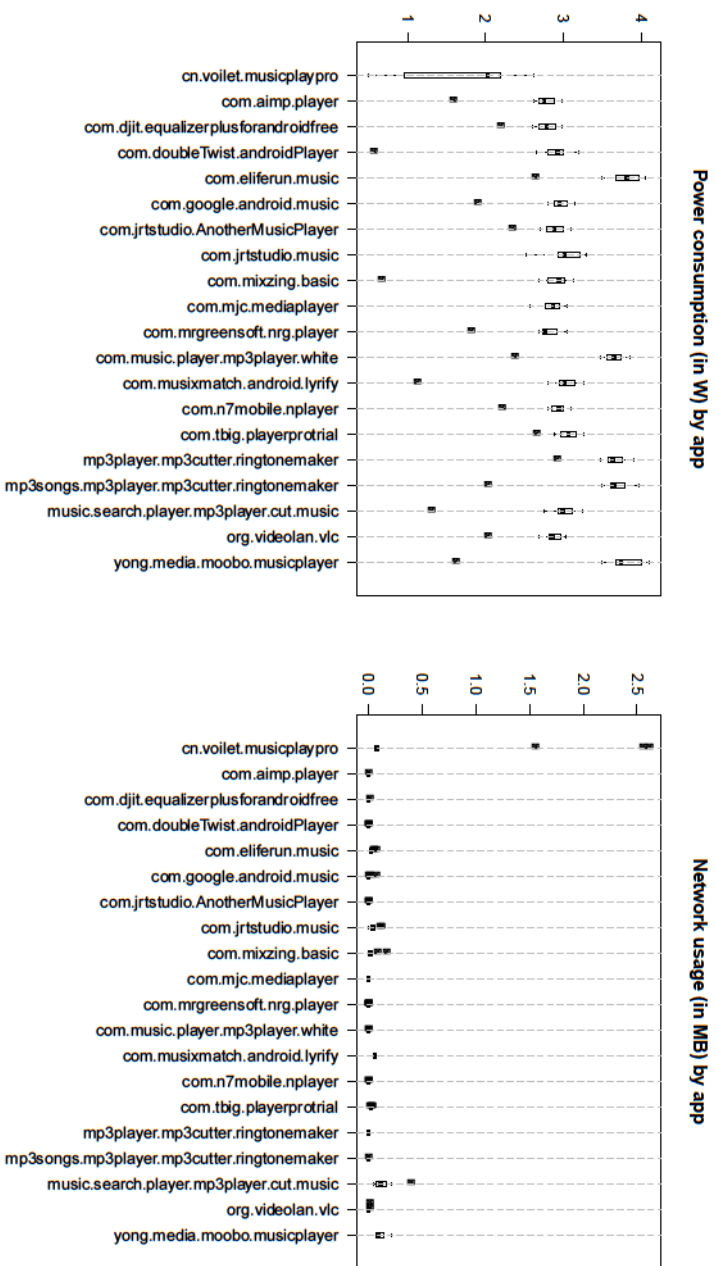


Figure 6: Average power consumption and network usage for apps in the *Music Players* category.

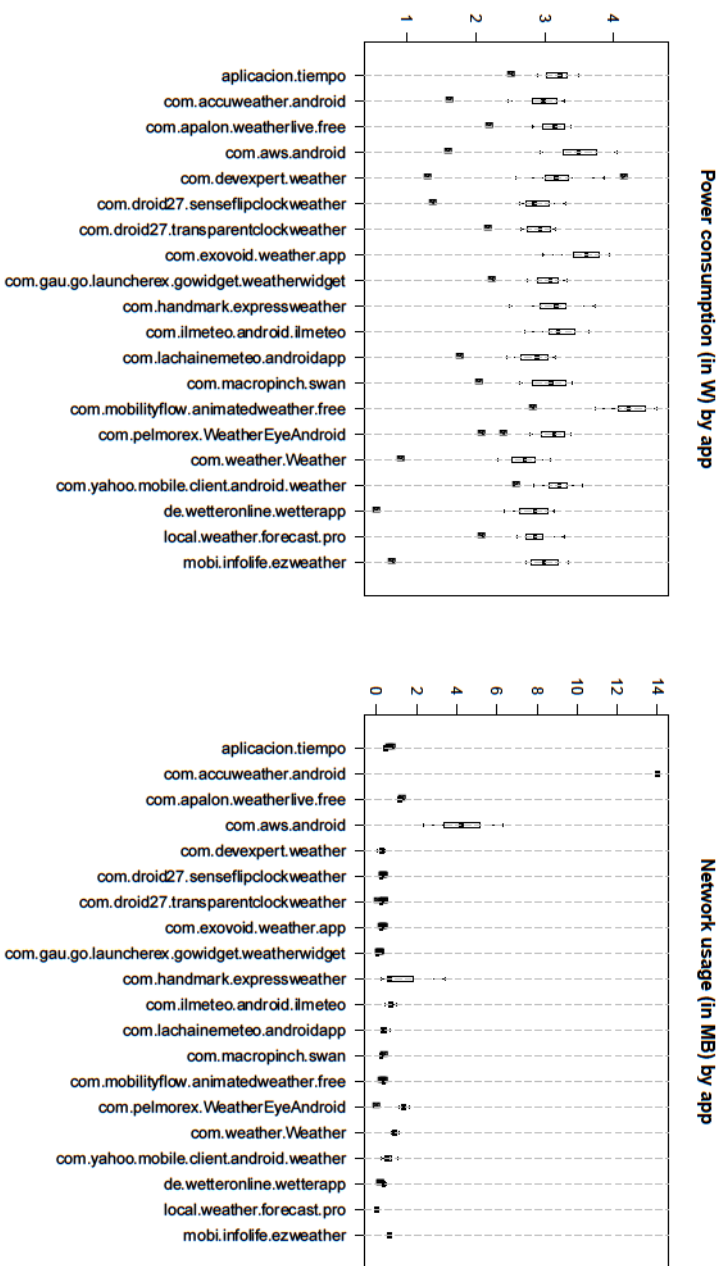


Figure 7: Average power consumption and network usage for apps in the *Weather* category.

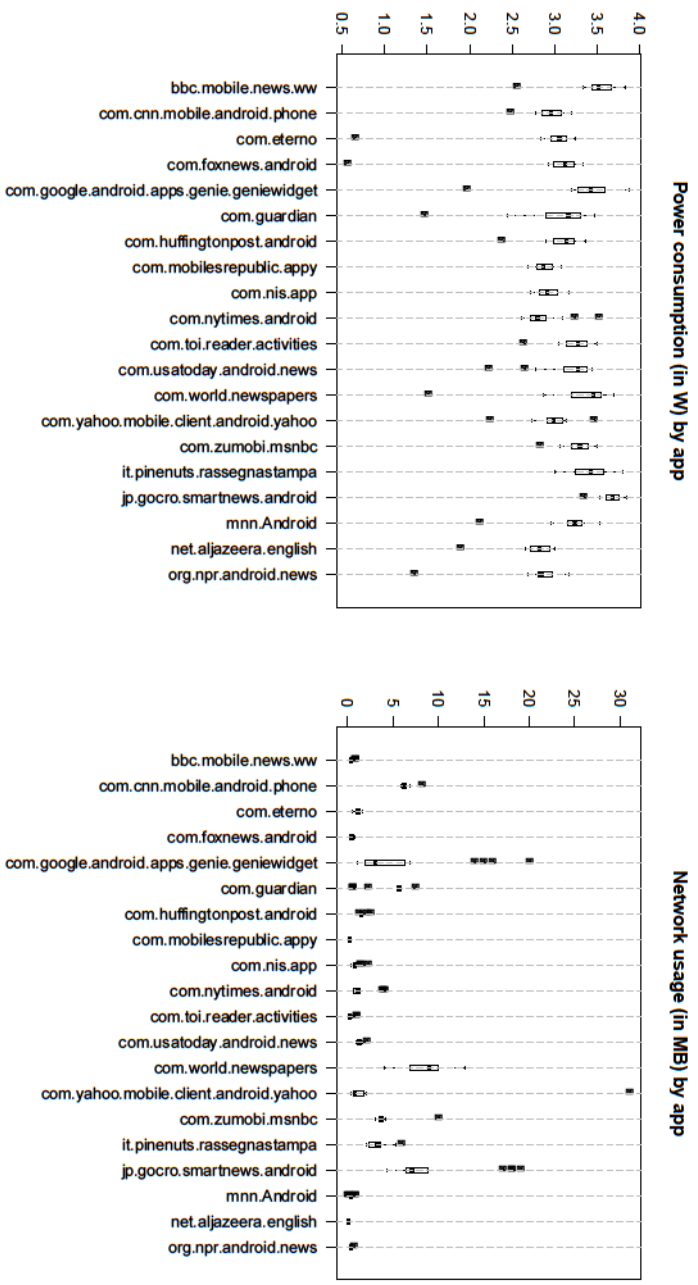


Figure 8: Average power consumption and network usage for apps in the *News* category.

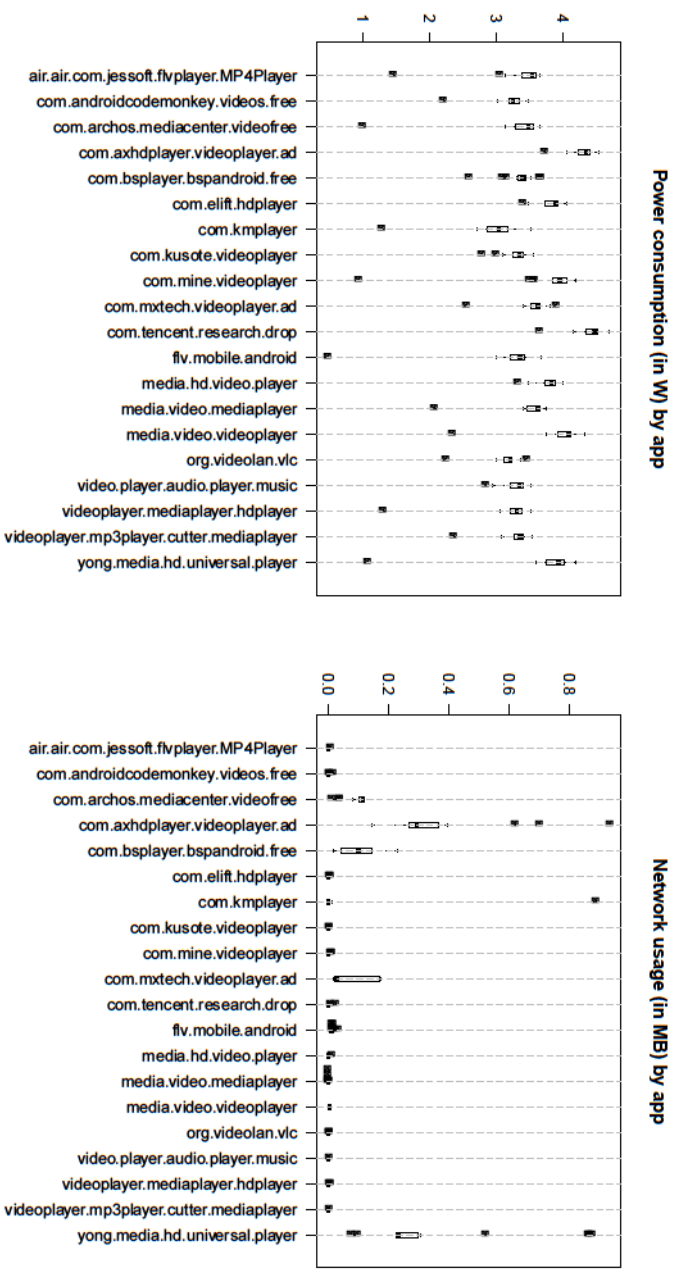


Figure 9: Average power consumption and network usage for apps in the *Video Players* category.

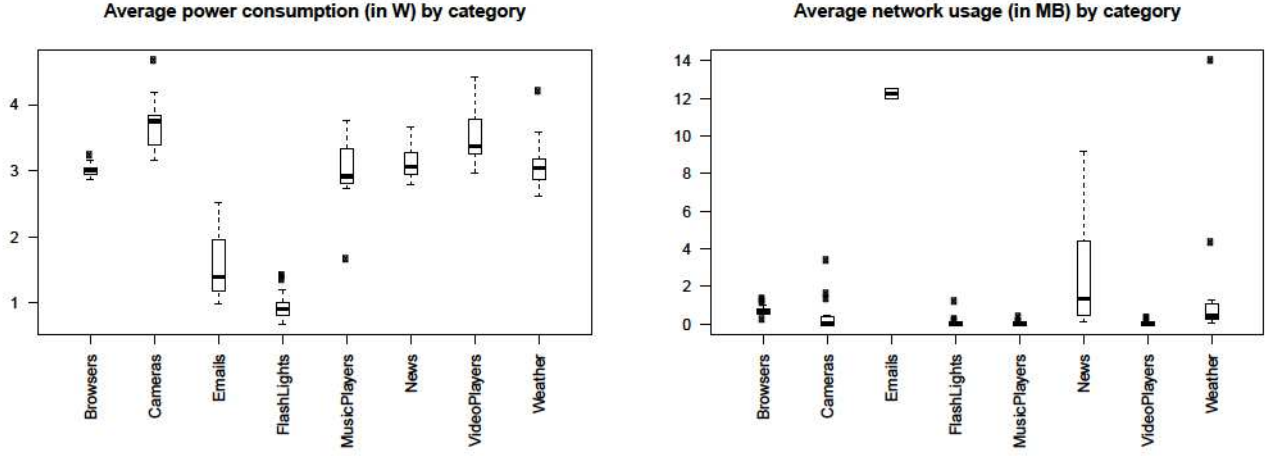


Figure 10: Average power consumption and average of network usage for applications in each category.

In order to analyze the tradeoff between power consumption, network usage, and rating, regarding the best applications in each category after the search space reduction, a set of plots, named *value paths*, is showed in Figure 11. In these plots, a bar is drawn for each objective and its height is defined by the minimum and maximum values of the corresponding objective in the corresponding category (considering all of the initial applications in the category and not only the non-dominated ones). A line is used to represent each application in the category specifying the value of each objective function.

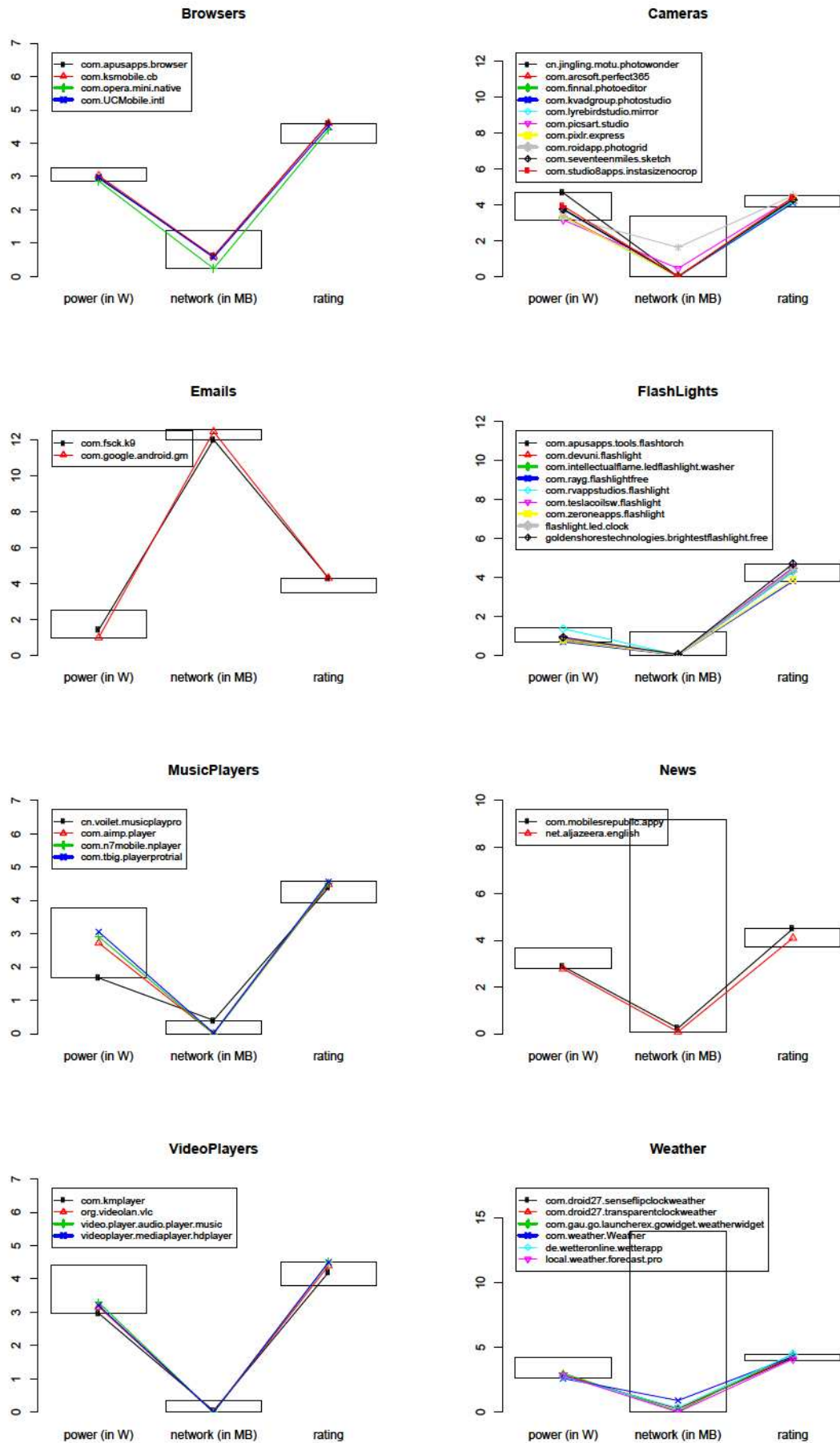


Figure 11: Power consumption, network usage, and rating, for best apps in each category.

In Figures 12-19, the comparison between Pareto optimal applications found by ADAGO, in the different categories, and the application chosen by the user are shown.

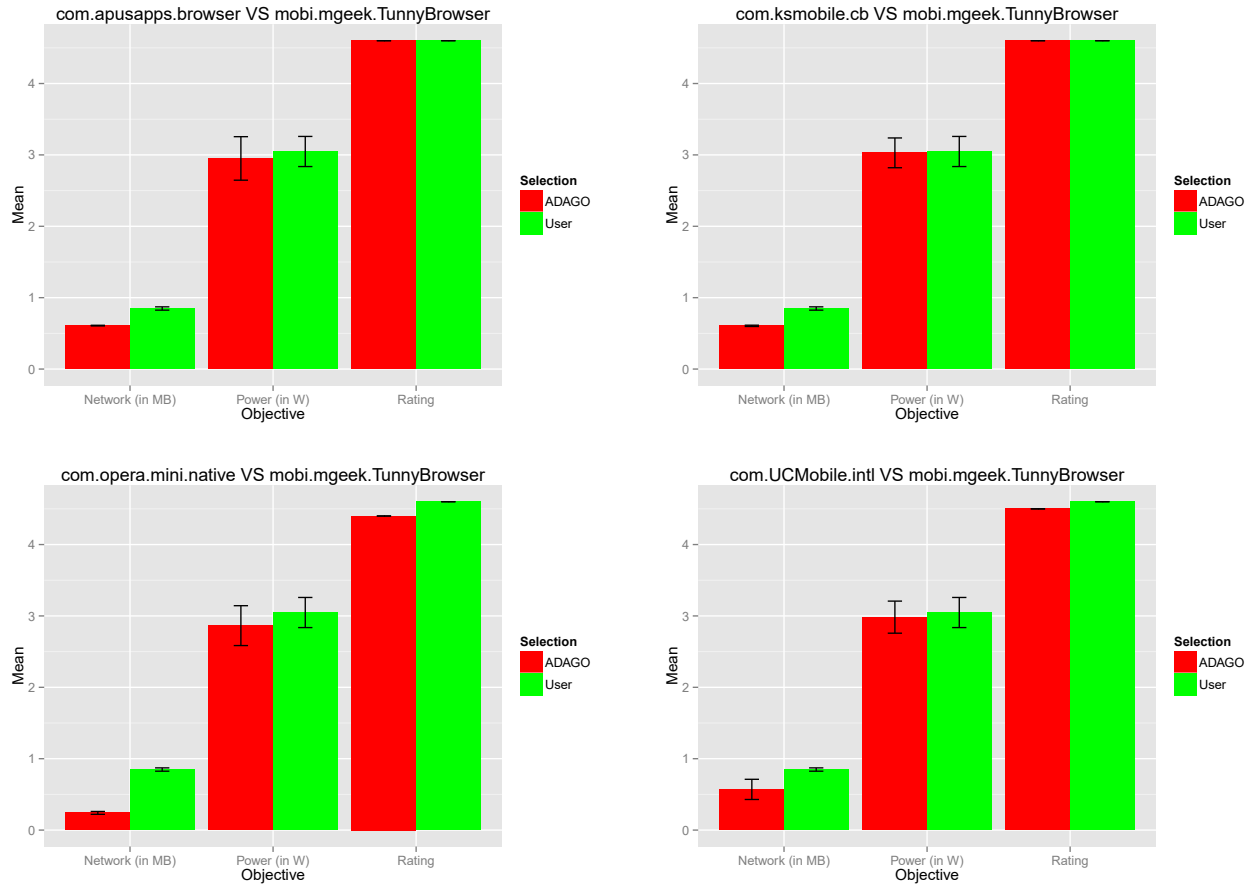


Figure 12: Comparing Pareto optimal apps found by the ADAGO, in the *Browsers* category, respect to the app chosen by the user.

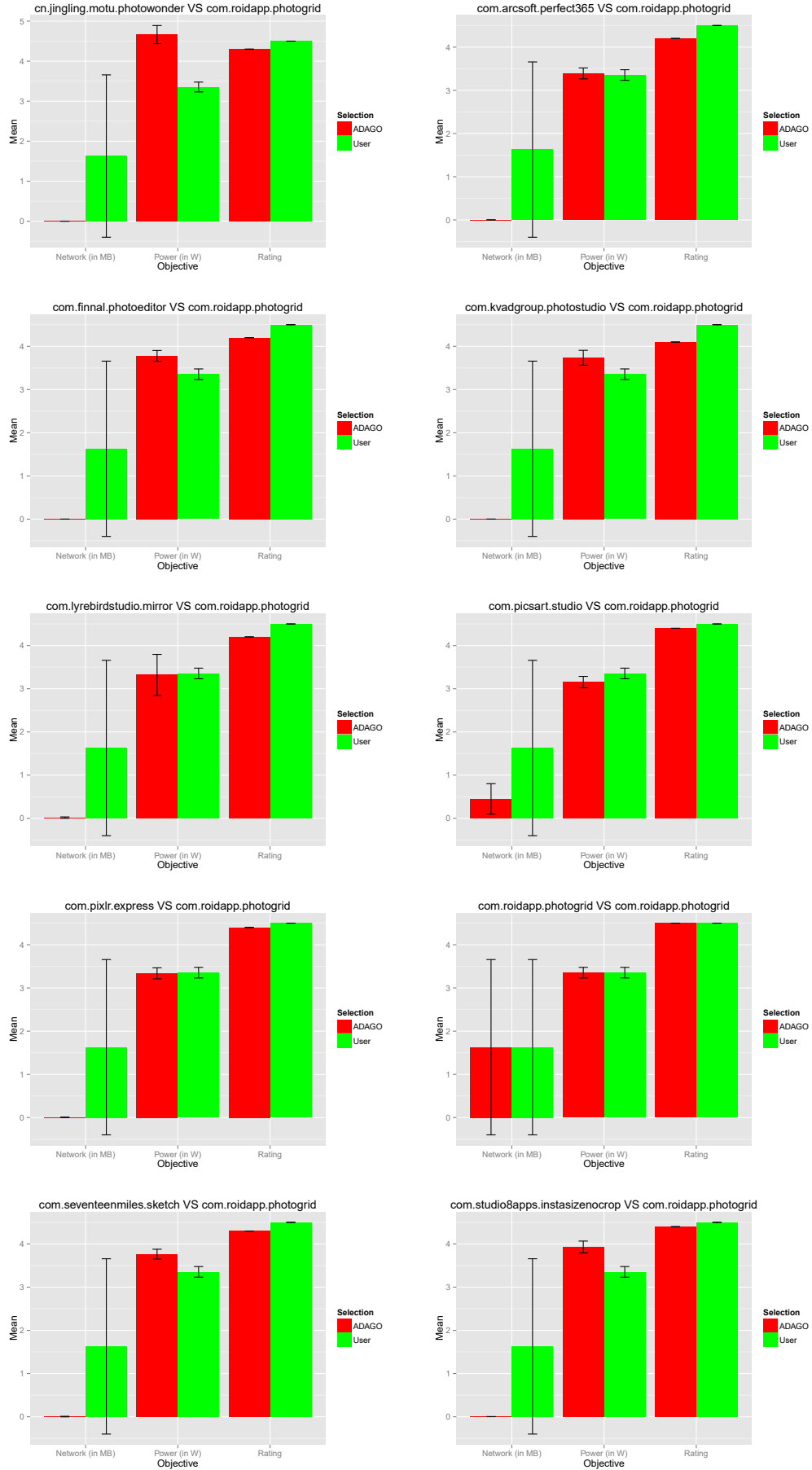


Figure 13: Comparing Pareto optimal apps found by the ADAGO, in the *Cameras* category, respect to the app chosen by the user.

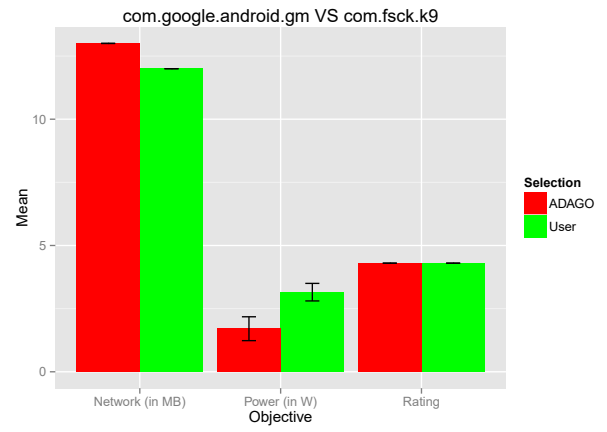
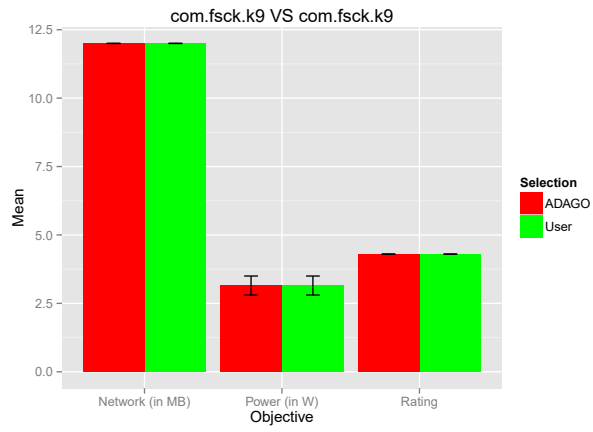


Figure 14: Comparing Pareto optimal apps found by the ADAGO, in the *Emails* category, respect to the app chosen by the user.

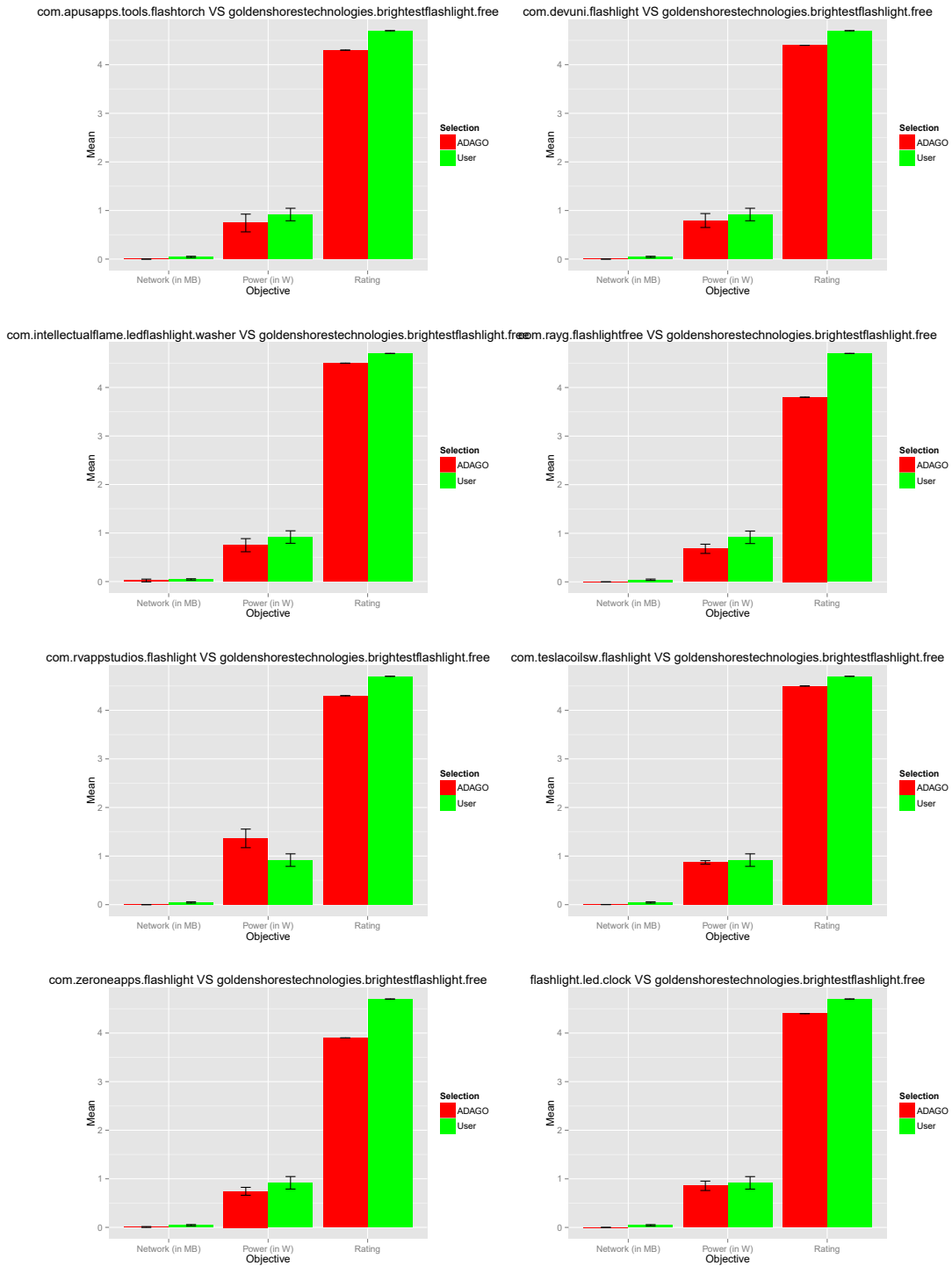


Figure 15: Comparing Pareto optimal apps found by the ADAGO, in the *Flash Lights* category, respect to the app chosen by the user.

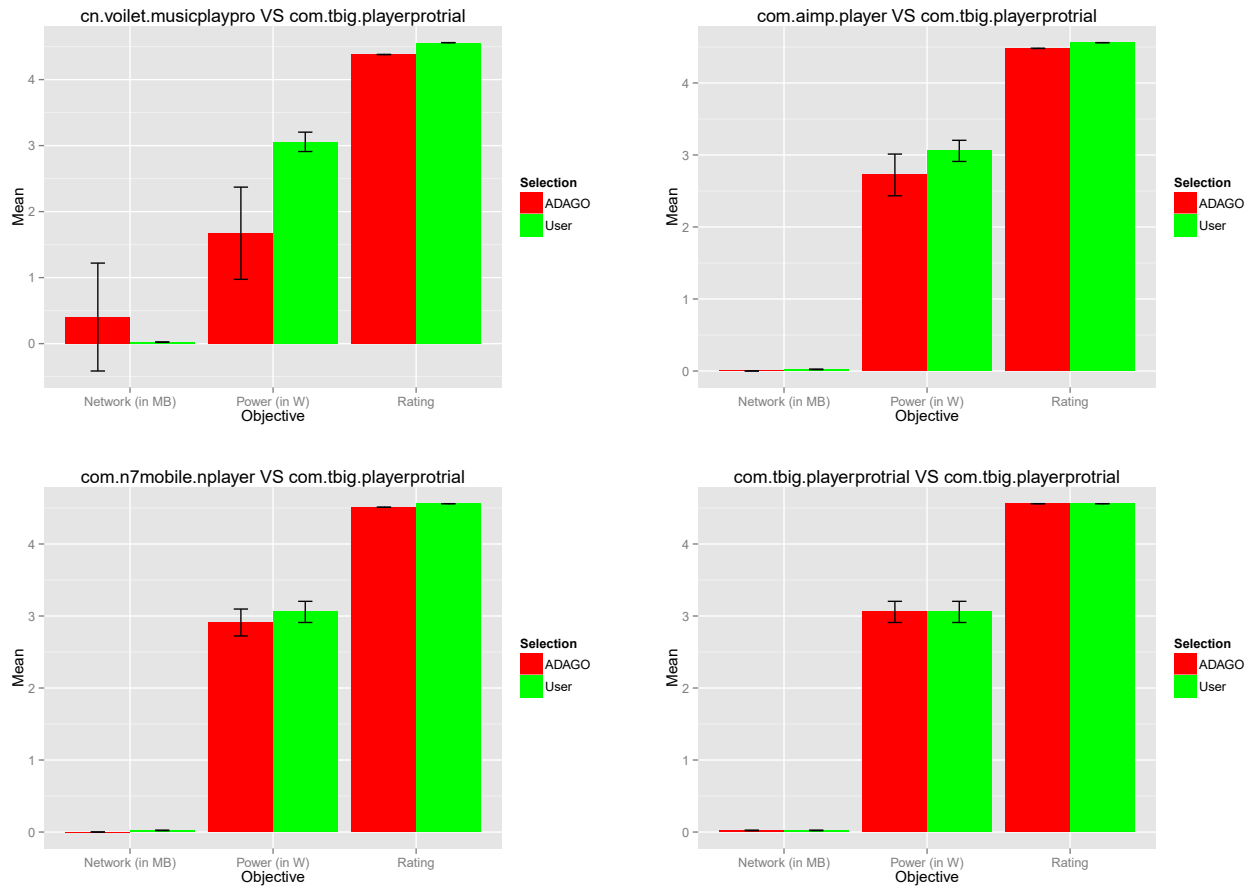


Figure 16: Comparing Pareto optimal apps found by the ADAGO, in the *Music Players* category, respect to the app chosen by the user.

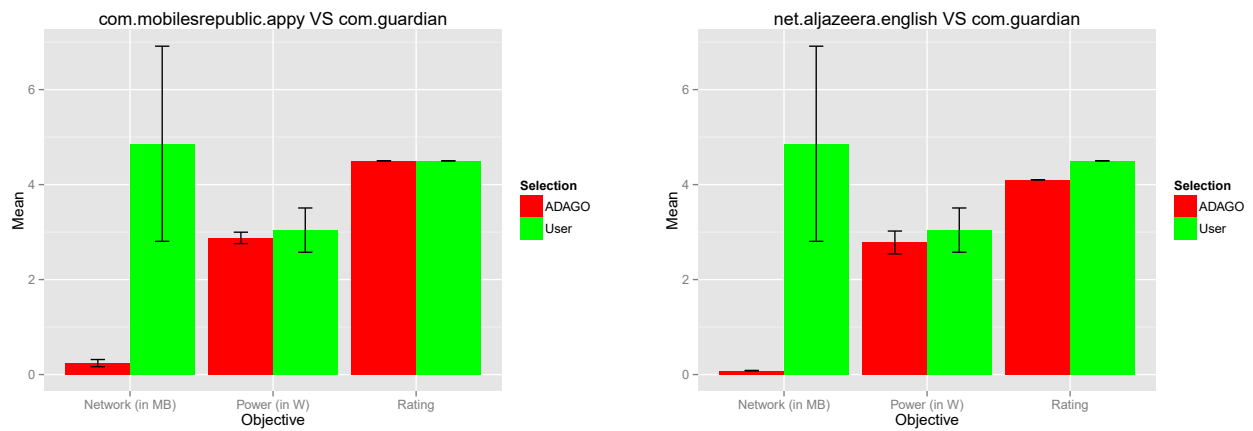


Figure 17: Comparing Pareto optimal apps found by the ADAGO, in the *News* category, respect to the app chosen by the user.

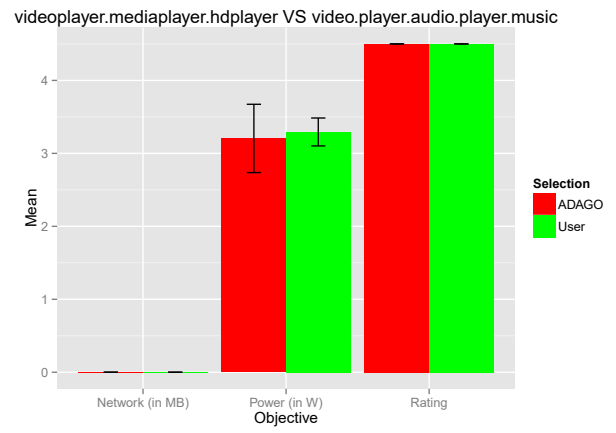
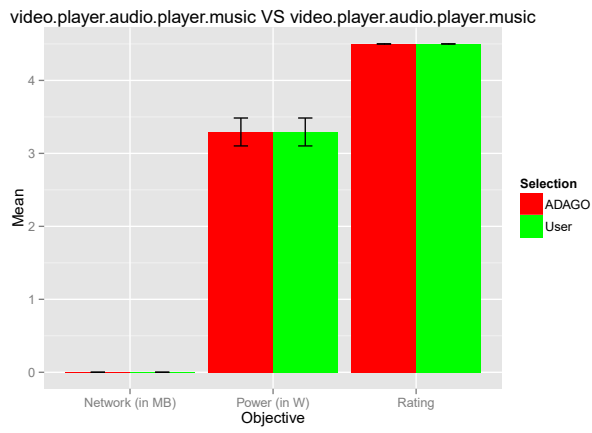
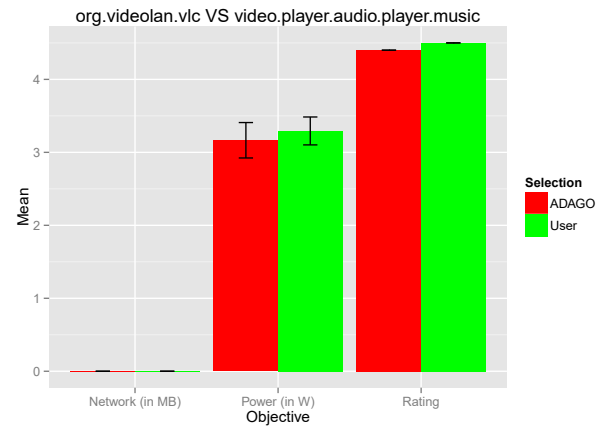
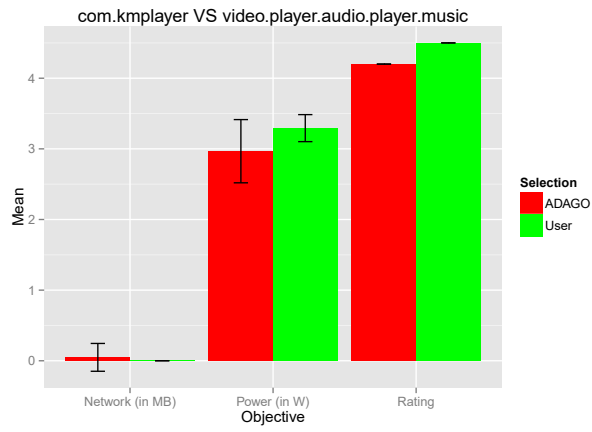


Figure 18: Comparing Pareto optimal apps found by the ADAGO, in the *Video Players* category, respect to the app chosen by the user.

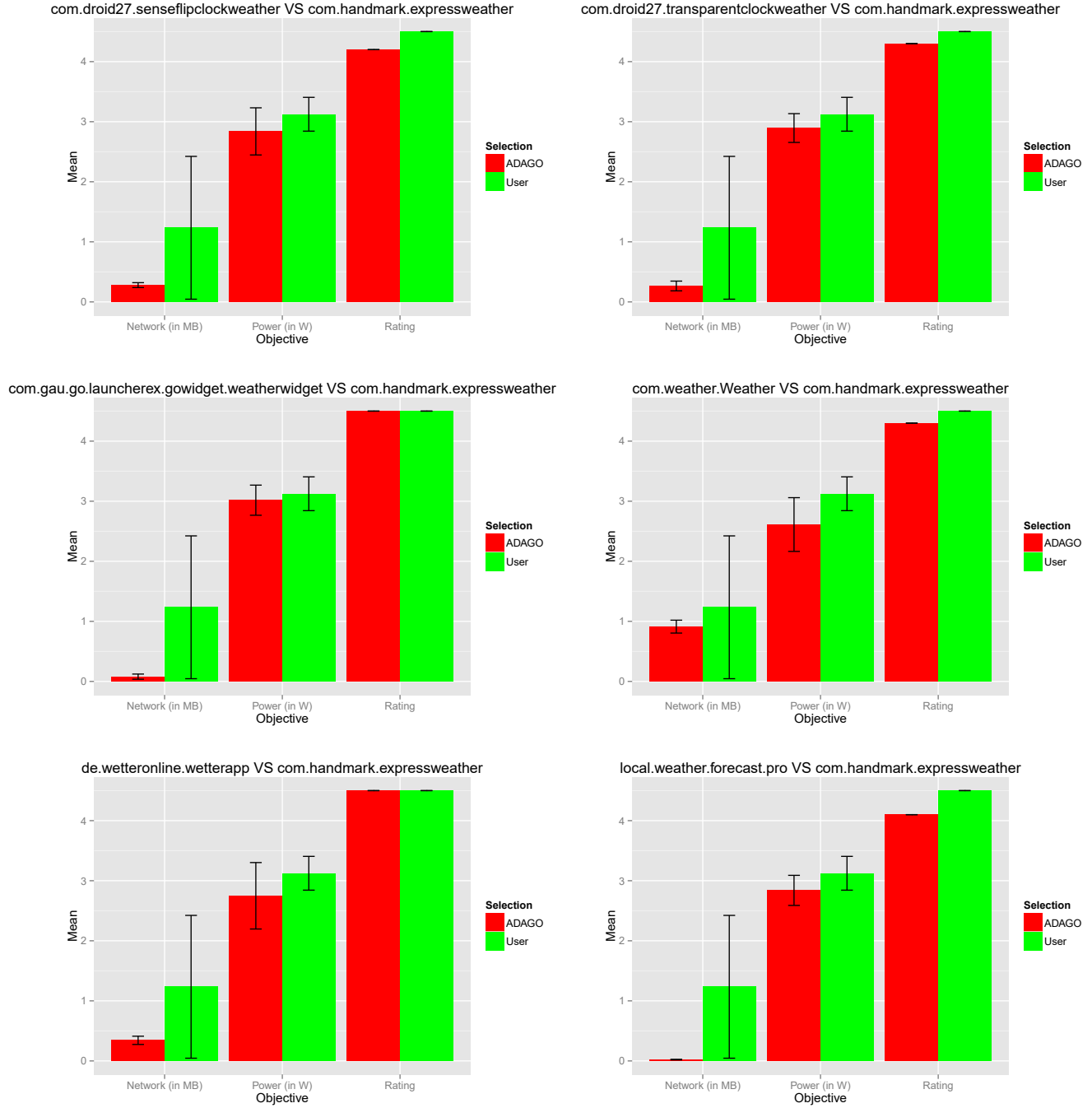


Figure 19: Comparing Pareto optimal apps found by the ADAGO, in the *Weather* category, respect to the app chosen by the user.

References

- [1] K. Deb. *Multi-objective Optimization using Evolutionary Algorithms*. Wiley, Chichester, 2001.
- [2] J.J. Durillo and A.J. Nebro. jMetal: A Java framework for multi-objective optimization. *Advances in Engineering Software*, 42:760–771, 2011.
- [3] Ruben Saborido, Giovanni Beltrame, Foutse Khomh, Enrique Alba, and Giulio Antoniol. Optimizing user experience in choosing android applications. In *Proceedings of the 23rd IEEE International Conference on Software Analysis, Evolution, and Reengineering (SANER)*, March 14-18, 2016.

L'École Polytechnique se spécialise dans la formation d'ingénieurs et la recherche en ingénierie depuis 1873



École Polytechnique de Montréal

**École affiliée à l'Université
de Montréal**

Campus de l'Université de Montréal
C.P. 6079, succ. Centre-ville
Montréal (Québec)
Canada H3C 3A7

www.polymtl.ca

