

# Mobile App User Choice Engineering using Behavioral Science Models

Merkourios Karaliopoulos and Iordanis Koutsopoulos  
 Department of Informatics  
 Athens University of Economics and Business, Greece  
 {mkaralio, jordan}@aueb.gr

**Abstract**—When interacting with mobile apps, users need to take decisions and make certain choices out of a set of alternative ones offered by the app. We introduce optimization problems through which we engineer the choices presented to users so that they are nudged towards decisions that lead to better outcomes for them and for the app platform. User decision-making rules are modeled by using principles from behavioral science and machine learning. Such instances arise in (i) mobile crowdsensing campaigns, where tasks are assigned to users through the app, and the goal is to optimize the quality of fulfilled tasks; (ii) smart-energy apps, where energy-saving recommendations are issued through the app, and the goal is to optimize energy savings; (iii) mobile advertising, where ads or offers are projected to the user, and the aim is to optimize revenue through user response to ads. Each user is modeled as a vector of feature values for a set of features. In an important class of decision-making models in behavioral science, the lexicographic fast-and-frugal-tree (FFT) heuristics, user decision emerges through a ranking of features that in turn gives rise to a decision tree. Having the incentive as a controllable feature that guides the user decision process, we study and characterize the complexity of the problem of allocating choices and incentives to users out of a limited budget. Numerical results indicate important performance gains when the incentive allocation policy adapts to user lexicographic choices.

**Index Terms**—User choice engineering, mobile apps, incentive allocation, behavioral science, decision trees.

## I. INTRODUCTION

Mobile applications (apps) and their interface with users are at the epicenter of active research that aims to make them more attractive and ultimately engaging for users. In a wide gamut of mobile apps, users need to take decisions and make certain choices out of a set of alternatives offered by the app. These choices crucially determine the performance of the app with respect to a certain objective that it aims to achieve. For example, a mobile crowdsensing (MCS) campaign app aims to assign tasks to users which require effort in terms of time, attention, device energy, user expertise or physical distance coverage to perform the task. Tasks could be *e.g.*, submitting measurements to create transportation or pollution maps, traveling across a city to capture photos from points of interest, submitting expert opinions, sharing data or advice for lifestyle and healthcare improvement, delivering parcels, and so on. A plausible aim of a mobile crowdsensing campaign

is to optimize task assignment to users so as to maximize the extracted social welfare of executed tasks.

In a smart-energy mobile app, recommendations prompt consumers towards energy-friendly habits [1]. Recommendations include *e.g.*, shifting (part of) their power load to times that avoid peak-time consumption and lead to savings in electricity generation cost. In yet another realm, that of mobile advertising, users are presented with various offers or ads through a mobile app. The goal is to assign the appropriate ad to a user at the appropriate time so as to maximize revenue stemming from positive user response to ads.

In this paper, we introduce optimization problems that engineer the *choices* presented to users so that they are nudged towards decisions that lead to better outcomes for them and for the app platform. Each user is modeled as a vector of feature values for a set of features that depends on the problem. We depart from the vast majority of literature works that model users as rational entities. Instead, we adopt principles from behavioral science and machine learning and leverage them in simple heuristic rules that presumably guide user choices. Specifically, we consider the class of *lexicographic* decision-tree heuristics for user decision making, where the key idea is to have an intrinsic ordering of available choices by users with respect to features, while decisions are further shaped by some threshold values. Depending on the ordering, a deterministic decision tree emerges whose branches depict the different possibilities for decisions.

In the instances above, users experience some cost or inconvenience while interacting with the app, and hence they need to be incentivized and convinced to engage and make choices that contribute to the objective of the app. However, users are heterogeneous in their choices and the motives behind them, and thus it is important for the app to understand their inherent hidden reasons behind these choices. Thus, it is critical to devise realistic models about users' choice-making behavior and in particular about their response to incentives which are a basic feature that may guide their decisions.

Consider for instance a mobile crowdsensing app that asks users to go to a certain place and do a task *e.g.*, take a photo, in return for some payment. Let us assume that there are two features that guide user decisions *i.e.*, the received payment (incentive) and the distance to be traveled to that place, and that there are two alternative choices offered. A user that prioritizes payment over distance will perform the

This work was supported by the European Commission H2020 Research Program under Grant Number 688768 netCommons (Network Infrastructure as Commons) and by AUEB-RC under the internal project "Original Scientific Publications".

task that offers substantially higher payment than the other. If payments do not differ so much so as to trigger a choice, the distance feature is activated, and the user will perform the task that is much closer than the other. If both tasks are at similar distance, the user may naturally perform none of the tasks or may choose one at random.

The contributions of this paper are summarized as follows.

- We devise decision-making models that use principles from behavioral science and machine learning. Specifically, we consider the class of lexicographic fast-and-frugal-tree (FFT) heuristics, where user decision emerges through a ranking of features that in turn gives rise to a decision tree. One of these features is the incentive provided to users.
- We introduce these models into the optimization problem of *choice engineering*, namely choice and incentive allocation to users out of a limited budget so as to maximize a welfare performance metric dictated by the app.

In section II we describe the model, and in section III we formulate the problem. In section IV we present results on model performance evaluation. We discuss state-of-the-art work in section V and conclude in section VI.

## II. MODEL

Let  $\mathcal{U}$  denote the set of users that interact with the mobile app, and let  $\mathcal{C}$  be the set of *choices*, also known as *alternatives* presented to users. For example, in mobile crowdsensing, the choices are the crowdsensing tasks that could benefit from user contributions, whereas in a smart-energy app the choices are more prudent energy consumption plans the consumers could subscribe to, and in mobile advertising, the choices are the ads or offer coupons presented to users.

Each choice  $i \in \mathcal{C}$  is modeled as a  $K$ -dimensional vector  $\mathbf{x}_i = (x_i(1), x_i(2), \dots, x_i(K))$ , where  $x_i(k)$  is the value of the  $k$ -th feature for choice  $i$ ,  $k = 1, \dots, K$ , and  $K$  is the number of features. For instance, a feature may be a payment or reward that the choice offers. Another feature may be the amount of effort that the choice requires, or another quantity that captures the user cost or inconvenience when making the specific choice. In the case of mobile crowdsensing, this can be the amount of time needed or the physical distance covered in order to contribute to an MCS task. In the smart grid case, it may refer to the inconvenience of changing daily habits and shifting the energy load in time; and so on. In general, there may be more features affecting a choice in a specific setting, *e.g.*, the crowdsensing task type or the ad category. For simplicity here, we assume that choices are determined by the values of two features, those representing the *incentive* and the incurred *cost* of a choice.

The incentives that come with the different choices are subject to budget constraints. Let  $b_i$  be the total amount of rewards (*e.g.*, payments, discounts, coupons) offered so as to attract users to choice  $i$ . For example, in mobile advertising, the different choices are the advertised stores, and each store places a certain budget on having its ads and offers displayed to users. In mobile crowdsensing, the budget is allocated to

each task by the task issuer to reward contributors. The mobile app then issues pop-up texts that read roughly as follows: “You can perform task 1 at distance  $d_1$ , *e.g.*, take photos of a place, for a reward  $p_1$ ; or you can perform task 2 at a distance  $d_2$ , *e.g.*, check and report on the quality of products of a store, for a reward  $p_2$ ”.

For each choice  $i \in \mathcal{C}$  and user  $u \in \mathcal{U}$ , let  $q_i^u$  be a weight factor that associates choice  $i$  and user  $u$ . For example, in mobile crowdsensing,  $q_i^u$  denotes the quality of contribution that user  $u$  can make to task  $i$ . In mobile advertising, it is  $q_i^u = q_i$  for all  $u$ , and this denotes the amount paid by store  $i$  to the platform per user visit. In smart-energy apps, it could be the energy savings if user  $u$  chooses energy plan  $i$ .

### A. Lexicographic user choice heuristics

Our work is inspired by the class of lexicographic heuristics for modeling user decision-making, whereby choice features are inspected in a user-specific order, and a choice is made based on the first feature that discriminates between the alternatives. Hence, the user does not exhaustively process all information at hand *i.e.*, the full feature set in order to make a choice. The accuracy of lexicographic heuristics in predicting human choices is comparable to that of more sophisticated or computationally demanding models such as those based on regression, neural networks, classification and regression trees or naive Bayes. These simple heuristics often perform better in predicting choices when compared to matching past choices. Further, each model outperforms the other under certain conditions, since each one is capable of exploiting different properties of the decision environment such as correlation between different features. This coupling between the structure of choice rules and the context within which decision takes place allows the former to efficiently exploit the latter, in what is called *ecological rationality* [2].

One instance of lexicographic heuristics is the fast-and-frugal-trees (FFTs) one [3]. FFTs are deterministic binary decision trees with at least one exit leaf at every level. In other words, for every feature that is inspected, at least one of its outcomes leads to a choice.

Consider the example of mobile advertising with features being the amount of offer and the distance to travel so as to retrieve the offer. Assume there are two alternative choices  $A, B$ , for each user  $u$ , specified by the reward-distance pairs  $(p_A^u, d_A^u)$  and  $(p_B^u, d_B^u)$ . Let  $\mathcal{U}_d, \mathcal{U}_p$  denote the subsets of users that place priority on the distance and the payment feature respectively, *i.e.*,  $\mathcal{U}_d \cap \mathcal{U}_p = \emptyset$  and  $\mathcal{U}_d \cup \mathcal{U}_p = \mathcal{U}$ .

Since features take continuous values, it makes sense to define for each user  $u$  a threshold  $\theta_p^u \geq 0$  for the payment and a threshold  $\theta_d^u \leq 0$  for the distance feature. The training phase of the model provides (i) the type of tree of each user  $u$ , depending on whether  $u \in \mathcal{U}_p$  or  $u \in \mathcal{U}_d$ , (ii) the threshold values,  $\theta_p^u, \theta_d^u$ . Consider a user  $u \in \mathcal{U}_p$  that prioritizes the offer reward over distance. If  $p_A^u - p_B^u > \theta_p^u$ , user  $u$  selects choice  $A$ , while if  $p_B^u - p_A^u > \theta_p^u$ , the user goes with choice  $B$ . If the difference in rewards is not large enough to warrant a choice, *i.e.*, if  $|p_A^u - p_B^u| \leq \theta_p^u$ , the user switches to the distance

feature. If  $d_A^u - d_B^u < \theta_d^u$ , the user selects choice  $A$ ; else, if  $d_B^u - d_A^u < \theta_d^u$ , she selects choice  $B$ . If the difference in the distances is not large enough to justify a choice with respect to the distance criterion, the user chooses none of the available choices. Fig. 1 depicts such a fast-and-frugal decision tree.

### III. USER CHOICE ENGINEERING AS AN OPTIMIZATION PROBLEM

The app owner or campaign designer aims at engineering the alternatives that will be made available to users by allocating options and incentives to them so that they make choices that are beneficial for the system objective as a whole. For instance, the app may recommend certain choices to users who prioritize based on exerted effort (*e.g.*, distance) and thus reduce monetary payments. These payments could instead be allocated to expert users who prioritize over the payment feature of the recommended choice. Thus, the budgets of the various choices can be managed more efficiently to attract more and better users. As mentioned above, for illustrative reasons we consider two features, the cost (*e.g.*, the distance), and the reward (*e.g.*, the payment) which is controllable. We assume that *two choices* are presented each time to a user.

#### A. Problem formulation

We need to allocate choices and incentives to users so as to maximize the total quality of user-selected choices, subject to a budget constraint for each choice. Let  $\mathcal{P} \subseteq \mathcal{C} \times \mathcal{C}$  denote the set of available pairs of choices  $(i, j)$  for assignment, with  $i \neq j$ . For each choice pair  $(i, j) \in \mathcal{P}$ , let the binary variable  $y_{(i,j)}^u = 1$  if the choice pair  $(i, j)$  is assigned to user  $u$ , and 0 otherwise. Let  $\mathbf{y}^u = (y_{(i,j)}^u : (i, j) \in \mathcal{P})$  be the 0-1 vector of choice pair allocations to user  $u$ , and let  $\mathbf{y} = (\mathbf{y}^u : u \in \mathcal{U})$  be the collective choice allocation policy to users.

When a choice pair  $(i, j)$  is allocated to user  $u$ , the app should decide whether to make one of the two choices clearly preferable to the other in terms of incentive payment. Formally, let variable  $z_i^u = 1$  when choice  $i$  is made clearly preferable to the other one, *i.e.*, when the incentives of choices  $i, j$  satisfy  $p_i^u - p_j^u > \theta_p^u$ . Due to limited budget, the incentive for a choice should be the minimum possible so that the user will make that choice. When  $z_i^u = 1$ , the minimum payment for choice  $i$  is  $\theta_p^u + \epsilon$ , where  $\epsilon$  is a small fixed amount of payment;  $\theta_p^u$  is the incentive difference between the two choices. Note that it is  $z_i^u = 0$  when  $p_i^u - p_j^u \leq \theta_p^u$ , and  $z_j^u = 1$  when  $p_j^u - p_i^u > \theta_p^u$ . Also, it is  $z_i^u + z_j^u \leq 1$ , thus we may also choose  $z_i^u = z_j^u = 0$ , *i.e.*, not make any of the choices clearly preferable to the other in the payment. Let  $\mathbf{z}^u = (z_i^u : i \in \mathcal{C})$  be the incentive allocation policy for user  $u$  that determines whether or not we make a choice with much higher incentive than the other. Let  $\mathbf{z} = (\mathbf{z}^u : u \in \mathcal{U})$ .

A user  $u \in \mathcal{U}_p$  will pick choice  $i$  if  $z_i^u = 1$ , and the payment will be  $p_i^u = \theta_p^u + \epsilon$ . If  $z_i^u = z_j^u = 0$ , then  $u$  will resort to the distance feature to make a choice. If the pair of presented choices  $(i, j)$  is such that  $d_i^u - d_j^u < \theta_d^u$ , user  $u$  will pick choice  $i$ . If  $d_j^u - d_i^u < \theta_d^u$ , the user will pick choice  $j$ . The payment will be an amount  $\delta$  that is a priori determined by the

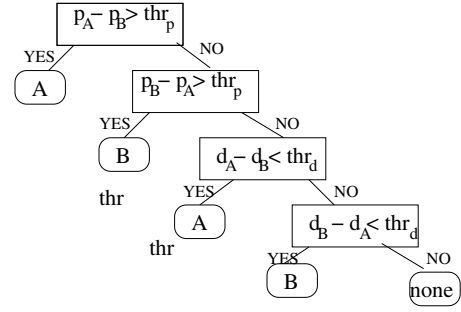


Fig. 1. A Fast and Frugal Tree (FFT) applied on two alternative choices with two decision features for a user in  $\mathcal{U}_p$ . Parameters  $thr_p > 0$  and  $thr_d < 0$  are the payment and distance thresholds, namely the minimum differences in the values of these features that render one choice preferable over the other.

platform. We define indicator parameters  $I_{ij}^u$  and  $I_{ji}^u \in \{0, 1\}$  as follows: when  $d_i^u - d_j^u \leq \theta_d^u$ , it is  $I_{ij}^u = 1$ , else  $I_{ij}^u = 0$ . When  $d_j^u - d_i^u \leq \theta_d^u$ , it is  $I_{ji}^u = 1$ , else  $I_{ji}^u = 0$ . Clearly, it is  $I_{ij}^u + I_{ji}^u \leq 1$ . If  $I_{ij}^u = I_{ji}^u = 0$ , the user will select none of the choices. On the other hand, a user  $u \in \mathcal{U}_d$  will pick choice  $i$  if  $I_{ij}^u = 1$  and choice  $j$  if  $I_{ji}^u = 1$ ; the payment will again be  $\delta$ . If  $I_{ij}^u = I_{ji}^u = 0$ , user  $u$  will use the payment feature to decide and will pick choice  $i$  or  $j$ , depending on whether  $z_i^u$  or  $z_j^u = 1$ , thus getting a payment of  $\theta_p^u + \epsilon$ .

The objective is to find the choice and incentive assignment policies  $\mathbf{y}^u, \mathbf{z}^u$  for each user  $u$  so as to maximize the total quality of selected choices by users, subject to a budget constraint for each choice. The statements above are quantified by defining functions  $\alpha_p^u(\cdot), \beta_p^u(\cdot)$  for each user  $u \in \mathcal{U}_p$  about quality of selected choices and budget spent respectively,

$$\alpha_p^u(z_i^u, z_j^u) = q_i^u z_i^u + q_j^u z_j^u + (I_{ij}^u q_i^u + I_{ji}^u q_j^u)(1 - z_i^u - z_j^u),$$

$$\beta_p^u(z_i^u, z_j^u) = (\theta_p^u + \epsilon)z_i^u + I_{ij}^u \delta(1 - z_i^u - z_j^u),$$

and functions  $\alpha_d^u(\cdot), \beta_d^u(\cdot)$  for each user  $u \in \mathcal{U}_d$ ,

$$\alpha_d^u(z_i^u, z_j^u) = I_{ij}^u q_i^u + I_{ji}^u q_j^u + (1 - I_{ij}^u - I_{ji}^u)(q_i^u z_i^u + q_j^u z_j^u),$$

$$\beta_d^u(z_i^u) = I_{ij}^u \delta + (1 - I_{ij}^u - I_{ji}^u)(\theta_p^u + \epsilon)z_i^u.$$

Then, the objective is written as

$$\max_{\mathbf{y}, \mathbf{z}} \sum_{u \in \mathcal{U}_p} \sum_{(i,j) \in \mathcal{P}} \alpha_p^u(z_i^u, z_j^u) y_{(i,j)}^u + \sum_{u \in \mathcal{U}_d} \sum_{(i,j) \in \mathcal{P}} \alpha_d^u(z_i^u, z_j^u) y_{(i,j)}^u \quad (1)$$

subject to a budget constraint

$$\sum_{u \in \mathcal{U}_p} \beta_p^u(z_i^u, z_j^u) y_{(i,j)}^u + \sum_{u \in \mathcal{U}_d} \beta_d^u(z_i^u) y_{(i,j)}^u \leq b_i, \quad \forall i \in \mathcal{C}, \quad (2)$$

$$z_i^u + z_j^u \leq 1, \quad \forall u \in \mathcal{U}, \quad \forall (i, j) \in \mathcal{P}, \quad (3)$$

$$\sum_{(i,j) \in \mathcal{P}} y_{(i,j)}^u \leq 1, \quad \forall u \in \mathcal{U}, \quad (4)$$

where constraint (4) says that at most one pair of choices should be allocated to each user. We refer to problem (1)-(4) as problem **(P1)**. Even for fixed values of variables  $\mathbf{z}$ , **(P1)** is an instance of the Generalized Assignment Problem (GAP), which is known to be NP-Hard. Exact algorithms for

small instances and approximation solutions for larger problem instances are described in [4].

*Remark 1.* In the model above, users choose between two alternatives. A simpler scenario would be that a *single* offer  $i$  is made to a user  $u$  by the app, and she decides to take it or leave it. Again, two thresholds  $\gamma_p^u$ ,  $\gamma_d^u$  dictate  $u$ 's decision. If  $u \in \mathcal{U}_p$ , the user first checks the incentive value  $p_i^u$  of offer  $i$ . If  $p_i^u > \gamma_p^u$ , the user adopts it, otherwise she checks the distance feature. If  $d_i^u < \gamma_d^u$ , she takes the offer else not. A similar line is followed for a user  $u \in \mathcal{U}_d$ . The decision variables are  $\{y_i^u\}$  for allocating single choices  $i$  to user  $u$  and  $\{z_i^u\}$  for deciding whether the choice incentive  $p_i^u$  will be made greater than  $\gamma_p^u$  or not, with  $u \in \mathcal{U}$ ,  $i \in \mathcal{C}$ . The problem formulation is a simpler variant of (P1), and we refer to that as (P1').

*Remark 2.* When choices are not distinguishable from each other in terms of the second tested feature, we assumed that a user selects none of the choices (*e.g.*, see right-most leaf in the FFT in Fig. 1). Another model might assume that the user picks *any* of the two choices at random.

#### IV. EVALUATION - NUMERICAL RESULTS

In this section, we evaluate the achievable performance gain when the incentive allocation process explicitly accounts for lexicographic decision-making. For the sake of concreteness, we carry out this evaluation in the context of mobile crowdsensing (MCS); the methodology however, readily generalizes to the other applications mentioned in sections I and II.

##### A. Methodology

We consider a set  $\mathcal{C}$  of  $C$  MCS tasks spread across a rectangular area of  $L \times L$ . We consider  $N$  users who roam across the area of interest and interact with an app that recommends MCS tasks to them. Users who want to make contributions to tasks need to travel to these physical locations. Each MCS task  $i \in \mathcal{C}$  comes with a budget  $b_i$  to reward user contributions. A user  $u$  presents distinct skills  $q_i^u$  for each task  $i$ . At any point in time, users stand at different distances from the  $C$  ongoing tasks, and the mission of the app is to optimize task recommendations and offered incentive payments to maximize the aggregate quality of user contributions.

Drawing parallels to the model in section II, the cost of alternative (*i.e.*, task)  $i$  for user  $u$  relates to its physical distance  $d_i^u$ , and the incentive accompanying a recommendation is the monetary reward  $p_i^u$  for contributing to the task. The choices of each MCS user (*i.e.*, accept the recommendation or not) are described by a different FFT, and the task assignment problem for given user locations is the single-offer problem (P1') described in Remark 1 above.

We generate random user locations and user profiles, each profile consisting in the type of tree and the two thresholds,  $\gamma_p^u$  and  $\gamma_d^u$  that drive user decisions. We then solve (P1') with the approximation algorithm described in [4] to compute the aggregate expected quality of contributions. This solution is compared to alternative heuristic rules that determine the recommended tasks and offered payments without accounting for the lexicographic structure in user decision making. Hence,

recommendations may be issued to a user for the task that lies closest to her or for the one she is most skilled for; while the task budget is split either equally among users that get recommendations for it or in proportion to their skills for it. The four combinations, abbreviated as CLOSE-EQ, CLOSE-PROP, SKILL-EQ, and SKILL-PROP, are summarized in Table I. The recommended tasks and offered payments in each one of the four cases are then processed by the decision trees profiling each user to determine whether the user will accept the offer or not.

##### B. Numerical results

Tables II and III report the expected aggregate quality achieved with a task recommendation and payment allocation policy that explicitly accounts for lexicographic choices by users (GAP column). The numbers correspond to averages over 50 simulation runs together with their standard variation. In the same tables, we report the expected aggregate quality under each of the four alternative heuristic policies.

In all instances, the achievable performance gain when the policy adapts to lexicographic choices is in the order of 1.5-2.2 times the one under the simpler policies. These figures persist over a wide range of values for the number of users, MCS tasks, budget distribution per MCS task, and payment threshold values, a subset of which is shown here due to space constraints.

#### V. RELATED WORK

The class of fast-and-frugal heuristics originated from Gigerenzer *et.al* [5] as an effort to develop simple heuristics which capture the fact that humans do not use all available information when deciding. These heuristics can perform better than many complex methods that have their sources in statistics or artificial intelligence [2]. Another similar class of heuristics are the lexicographic (LEX) ones [6] which also parse features of different choices in a certain order and terminate upon the first feature that is different among alternative tasks; they then make the choice for which the selected feature is best. While LEX decides in favor of a choice even if a feature for this choice is only a little better than that for another choice, another variant, Lexicographic semiorder (*LEX-Semi*) [7] suggests that the difference in a certain feature should exceed a predetermined threshold. The equivalent of the LEX class for binary-valued features is the Deterministic Elimination by Aspects (DEBA) heuristic [8]. DEBA parses features sequentially and eliminates choices that do not address or have inadequate value in that feature. For continuous-valued features, the adequacy of a feature is assessed by comparing its value to the median feature value over all tasks.

Mobile user profiling has been considered previously in the literature in various contexts. In mobile crowdsensing where users are assigned tasks and need to select some to fulfill, our previous work [9] sought to predict the likelihood that a user fulfills a task through logistic-regression models and binary classification. We formulated and characterized the complexity of task assignment with the objective to maximize total

TABLE I  
HEURISTIC RULES FOR RECOMMENDING TASKS AND OFFERING REWARDS TO MCS USERS.

	Recommend the closest task	Recommend the task the user is most skilled for
Split the task budget equally	CLOSE-EQ	SKILL-EQ
Split the task budget in proportion to user skills	CLOSE-PROP	SKILL-PROP

TABLE II  
COMPARISON OF TASK RECOMMENDATION AND PAYMENT OFFER STRATEGIES.  $C=25$ ,  $L=1\text{km}$ ,  $\gamma_p^u \sim \mathcal{U}[0.5, 1.5]$ ,  $\gamma_d^u \sim \mathcal{U}[100, 500]$

	GAP	CLOSE-EQ	CLOSE-PROP	SKILL-EQ	SKILL-PROP
$N=200$ , $b_i \sim \mathcal{U}[10, 20]$	172.67± 2.2	85.03± 7.3	88.24± 6.5	100.26± 6.37	111.64± 4.17
$N=100$ , $b_i \sim \mathcal{U}[10, 20]$	87± 1.55	47.9± 3.18	47.39± 2.76	59.47± 3.07	59.93± 2.97
$N=200$ , $b_i \sim \mathcal{U}[15, 25]$	172.65± 2.36	93.05± 5.51	93.59± 4.91	110.27± 4.67	112.22± 4.42

TABLE III  
COMPARISON OF TASK RECOMMENDATION AND PAYMENT OFFER STRATEGIES.  $C=30$ ,  $L=1\text{km}$ ,  $\gamma_p^u \sim \mathcal{U}[1, 2]$ ,  $\gamma_d^u \sim \mathcal{U}[100, 500]$

	GAP	CLOSE-EQ	CLOSE-PROP	SKILL-EQ	SKILL-PROP
$N=200$ , $b_i \sim \mathcal{U}[10, 20]$	179.55± 2.19	80.17± 5.84	85.14± 5.4	101.44± 7.8	125.86± 4.49
$N=100$ , $b_i \sim \mathcal{U}[10, 20]$	89.02± 1.59	46.89± 3.99	46.34± 3.89	56.49± 3.11	57.99± 2.93
$N=200$ , $b_i \sim \mathcal{U}[15, 25]$	179.72± 1.38	90.56± 4.81	92.18± 4.521	120.16± 5.92	127.91± 3.96

expected task quality. In [10], we considered the allocation of incentives to users for which the likelihood to perform a task was captured through a continuous-valued willingness function that was concave in provisioned incentives. In that work, constraints on exerted effort such as total distance to perform a task were considered. The paper [11] includes a study of different threads on individual and social strategic decision making and reasoning under certainty or uncertainty.

Behavioral concepts touch upon the area of product marketing as well. Conjoint analysis aims to determine the subset of features that is most influential for choice of products. The approach entails a controlled set of potential products or services shown to users for comparison. Conjoint analysis has been used for assessing privacy in social apps [12] and for predicting user preferences in online platforms [13].

## VI. CONCLUSION

Concepts from cognitive and behavioral science remain, to the best of our understanding, largely unexploited by most of the (wireless) networking community. Our approach is an important step in user modeling that departs from approaches that use utility functions *e.g.*, logarithmic ones, parametrized by a few parameters to distinguish among users. The models appear to have broader implications in various application areas where user-app interaction and decision making arise at individual or social level with strategic user interactions. Besides mobile crowdsensing, smart-energy apps and mobile advertising, the models have significant repercussions on how to engineer choices offered to users in recommender systems, online social networks, social media platforms, and more. The ultimate goal is to appropriately engineer choices so as to nudge users towards decisions that lead to better outcomes for user experience and for the service or platform welfare.

Our study relied on decision-making models inspired by FFTs and can be extended through sophisticated machine-learning techniques on FFT learning (*e.g.*, learning the various

thresholds that guide user choices) and learning uncertainty, as well as through other models from behavioral science. Feature selection techniques could be used to determine the subset of features that build user profiles and guide user choices. Another interesting and non-trivial extension concerns the presentation of more than two choices to users.

## REFERENCES

- [1] I. Koutsopoulos, T. G. Papaioannou, and V. Hatzis, "Modeling and optimization of the smart grid ecosystem," *Foundations and Trends in Networking*, vol. 10, no. 2-3, pp. 115316, 2016. [Online]. Available: <http://dx.doi.org/10.1561/13000000042>.
- [2] K. V. Katsikopoulos, "Psychological heuristics for making inferences: Definition, performance, and the emerging theory and practice," *Decision Analysis*, vol. 8, no. 1, pp. 1029, Mar. 2011.
- [3] L. Martignon, O. Vitouch, M. Takezawa, and M. R. Forster, "Naive and yet enlightened: From natural frequencies to fast and frugal decision trees," *Thinking: Psychological Perspectives on Reasoning, Judgment and Decision Making*, Ch. 10, pp. 189211, 2005.
- [4] S. Martello and P. Toth, Generalized assignment problems, in *Proc. Intern. Symp. on Algorithms and Computation, (ISAAC)*, 1992, pp. 351-369.
- [5] G. Gigerenzer and R. E. Selten, *Bounded Rationality: The Adaptive Toolbox*. MIT Press, 2001.
- [6] P. C. Fishburn, "Lexicographic orders, utilities and decision rules: A survey," *Management Science*, vol. 20, no. 11, pp. 14421471, 1974.
- [7] J. W. Payne, E. Johnson, and J. Bettman, *The Adaptive Decision Maker*, Cambridge University Press, 1993.
- [8] R. M. Hogarth and N. Karelaia, "Simple models for multiattribute choice with many alternatives: When it does and does not pay to face tradeoffs with binary attributes?" *Management Science*, vol. 51, no. 12, pp. 1860-1872, 2005.
- [9] M. Karaliopoulos, I. Koutsopoulos, and M. Titsias, "First learn then earn: optimizing mobile crowdsensing campaigns through data-driven user profiling," in *Proc. ACM MobiHoc*, 2016.
- [10] P. Micholia, M. Karaliopoulos, and I. Koutsopoulos, "Mobile crowdsensing incentives under participation uncertainty," in *Proc. of MSCC workshop (in ACM MobiHoc)*, 2016.
- [11] M. Karaliopoulos, L. Spiliopoulos, and I. Koutsopoulos, "Factoring the human decision-making limitations in mobile crowdsensing," in *Proc. IEEE/IFIP Wireless on-Demand Net. Sys. and Serv. Conf. (WONS)*, 2016.
- [12] Y. Pu and J. Grossklags, "Towards a model on the factors influencing social app users? valuation of interdependent privacy," in *Proc. on Privacy Enhancing Technologies*, 2016.
- [13] S. Nikou, H. Bouwman, and M. de Reuver, "A consumer perspective on mobile service platforms: A conjoint analysis approach," *Communications of the Association of Information Systems (CAIS)*, vol. 34, article 82, 2014.