Evaluating Mobile Music Experiences: Radio On-the-Go

Anupriya Ankolekar, Thomas Sandholm, and Louis lei Yu[1]

Gustavus Adolphus College, St Peter MN 56082, USA,
lyu@gac.edu,
WWW home page: http://homepages.gac.edu/~lyu/

Abstract. Music has become an accompaniment to everyday activities, such as shopping and navigating. Although people listen to music in a context-driven manner, music recommendation services typically ignore where a user is listening to the music. They also typically select music based on a single seed song, rather than ordering a users created playlists for the best user experience. The contributions of this paper are three-fold: (1) We present a survey of 15 DJs of college radio stations to identify their heuristics in creating playlists for radio shows. (2) We present an experimental study design to evaluate various scheduling (track ordering) strategies for mobile music consumption *in situ*, which is used to (3) conduct a field experiment that compares the user experience of three scheduling strategies (tempo, genre and location) against the gold standard of a playlist created by an experienced DJ[1].

Key words: user experience, mobile music consumption, music scheduling, experiment design

1 Introduction

Listening to music on-the-go has been a fundamental part of our culture ever since the introduction of the transistor radio in 1954. With the popularity of portable digital players, people began to use music as an accompaniment to everyday activities, such as shopping and navigating [7, 6]. DeNora [11] has described the way music has begun to serve a personal function for people: encouraging concentration during important tasks, reducing stress, providing mental preparation and even as a way of organizing people's memories of key events. Personally chosen collections of music are organized by people into playlists as a way of accessing these personal functions as and when needed.

Over the last decade, automatically generated playlists [5, 23, 26] through digital music recommendation services, such as Pandora[2], Last.fm[3], and Spo-

[1] This work was completed when Anupriya Ankolekar and Thomas Sandholm were both researchers, and Louis Lei Yu was a postdoctoral research fellow at Hewlett Packard Labs. The majority of the experiments were conducted during the summer of 2011. The authors are listed here in alphabetical order.

[2] pandora.com

[3] last.fm

tify[4], and music player services such as the iTunes Genius Mix and Google Play's instant mixes, have become popular. The Echo Nest [5] is a platform that offers personalized music selection as a service to music and radio providers.

Although automatic music selection on music streaming services has proven popular, there is relatively little literature on music scheduling[6] strategies. In current music selection approaches, the scheduling of tracks emerges from individual track recommendations, rather than being designed for a smooth listening experience the way radio show producers or DJs do. Furthermore, although people's music listening is often driven by their physical context, playlist generation methods rarely take this into account (with the notable exception of [24]).

In this paper, we address this research gap through three contributions: (1) We present the results of an informal survey of 15 DJs of college radio stations, to compile and examine the heuristics they use in creating radio shows. (2) We present an experimental study design to evaluate various scheduling strategies for mobile music consumption *in situ*. Using this experimental design, we present the results of a small-scale field experiment that compares the user experience of three music-feature based scheduling strategies against the gold standard of a playlist or *schedule* created by an experienced DJ.

The experiment has been designed to measure user experience without disrupting the users' experience of the flow of music, while mitigating the effect of users' musical preferences. We present the requirements for our music scheduling field experiment and develop an experiment design that fulfills these requirements. Using this experiment design and based on the heuristics used by DJs, we compared 3 scheduling strategies: *genre-based scheduling*, *tempo-based scheduling*, and *location-based scheduling* against an expert schedule in terms of user experience for people listening to music on-the-go.

The organization of the paper is as follows: We begin with describing the related literature on music consumption and on automatically generated playlists. We then present the results of our informal survey with DJs in Section 2.3 and derive scheduling strategies to be used in the experiment. In the following section, we describe the requirements of an experiment design for scheduling music for mobile consumption, and the actual field experiment design and analysis. In the Experiment section, we describe the schedulers used in the experiment and the experiment procedure. The results of the experiment are reported next, in terms of perceived user experience based on explicit user ratings as well as users' self-reported emotional response to the various scheduling conditions. The paper ends with a discussion of the results and the generality of the experiment design.

[4] `spotify.com`

[5] `echonest.com`

[6] in this context, meaning the order in which we choose to play songs

2 Related Work

2.1 Music Consumption: Mobile and Location-Aware

DeNora [11] has conducted seminal research on how people consciously use music in their everyday life to perform various functions, such as encouraging concentration during important tasks, reducing stress, providing mental preparation and even as a way of organizing people's memories of key events. The ubiquitous culture of listening to music on mobile devices has been described by Bull [7, 6], who has documented how people listen to music while carrying out outdoor activities, such as shopping or navigating. Several studies have examined mobile music consumption in urban areas [22] and by youth [18], and how culture has a significant impact on how music is consumed. Nettamo et al. [22], in particular, describe how playlists are used to filter and organize vast and diverse collections of music to suit certain moods or contexts of use.

Recently, researchers began to examine the use of audio, and in particular music, for navigation outdoors. Nemirovsky and Davenport [21] developed a system called GuideShoes that utilizes a custom mobile music player built into shoes with GPS to deliver musical cues for street navigation. Warren et. al [28] present the Ontrack system to adapt audio continuously to help users navigate to a destination. Finally Gaye et. al [13] provides a good survey of mobile music research and early attempts to use location-based features with music.

In the last couple of years, location-aware music recommendation has flourished, with services like Soundtracker[7] and Soundtracking[8], and apps like RjDj[9], to name a few. Musicians have created location-aware albums, such as Bluebrain's the National Mall and Central Park, which are musical albums meant to be heard within a particular location, where the music heard is affected by the user's path. Music has also been shown to be an effective way to guide people to certain points of interest [3].

Although people commonly create personal playlists for different contexts, such as listening to music on-the-go, no existing system examines how to enhance the user's experience of music and their location by better scheduling the existing tracks in a user's playlist.

2.2 Automatic Playlist Creation

There is a parallel body of literature on automatic playlist creation within the music information retrieval community. The methods developed typically rely on various kinds of features of the audio, e.g. the metadata (such as artist, genre etc.) and content features (such as amplitude, beats etc.) to define similarity between audio. Ragno et al. [26] describe a way of automatically infer similarities between songs based on derived measures such as artists, genre, pitch, and

[7] `soundtracker.fm`

[8] `soundtracking.com`

[9] `rjdj.me`

tempo. Several playlist creation methods use such similarity metrics to automatically generate playlists of similar songs[5, 23]. In addition, automatic playlist generation methods typically rely on some form of explicit user preferences, such as a search query [25], a seed song [19, 2], or user skipping behaviour [23, 5].

PATS [24] generates playlists that suit a particular context-of-use, i.e. the real world environment in which the music is heard (such as music for work). To create playlists, it uses a dynamic clustering method in which songs are grouped based on a weighted similarity of attributes. An alternative approach to playlist generation treats the problem of selecting relevant music for a user as a collaborative filtering problem [17] and attempts to help the user find new music that matches their taste profile. Flycast [16] is another system that uses collaborative filtering techniques to generate a playlist based on the request histories of the current listening audience. Although not a collaborative filtering system, CoCoA Radio [4] allows users to collaborate on creating playlists for certain themes.

Unlike these approaches, our focus is not on selection of music, rather on scheduling a given set of songs into coherent and pleasant-sounding segments. Although these problems are related, the scheduling problem is more challenging, because the set of music to order is significantly smaller. Furthermore, as we shall see in the next section, DJs create engaging playlists by creating a sense of progression or movement within a schedule, which goes beyond methods that simply choose the 'best next song'. Finally, besides [24], none of the systems evaluate the performance of these techniques in a mobile context.

2.3 DJ Techniques

In order to better understand how DJs select and sequence songs for their shows, we conducted an informal survey of 15 DJs from college radio stations in the U.S. and Canada[10]. In addition, we examined the on-air training manuals for several college radio stations to understand how radio DJs develop radio programs[11]. In the following, we summarize our findings on the techniques DJs use when creating their shows.

College radio DJs typically select and sequence music from a large collection of vinyls, CDs, cassettes and digital downloads to produce a show. Most radio slots are 1-2 hours long and consist of more than 15 songs. A rule of thumb that many DJs use to keep listeners engaged for this long is to break the show up into segments of 3-4 songs [8], keeping each segment to be "a maximum of three

[10] The radio stations are (1) CFRC 101.9 FM, Queens University Radio (http://cfrc.ca), (2) CFUV 101.9 FM, University of Victoria Radio (http://cfuv.uvic.ca), (3) KUSF 90.3 FM, University of San Francisco Radio (http://savekusf.org), (4) CFYT 106.9 FM, Dawson City Community Radio (http://cfyt.ca) and (5) WRHU 88.7 FM, Hofstra University Radio (http://www.hofstra.edu/Academics/Colleges/SOC/WRHU).

[11] Unlike commercial radio stations whose playlists have been automatically generated [27] to get the best possible ratings [12], college radio stations still tend to have DJs who choose and schedule the music for their own shows.

songs or 15 minutes, which ever comes first." [9]. In between these segments, DJs might talk on air or play promos, announcements or commercials. These breaks provide some change to the listeners and provide natural points at which the DJs can change the pace of the program, in some sense, clearing the slate of one segment to start afresh in the next.

When selecting music, DJs often try to find a coherent theme to tie songs together. E.g. a DJ may dedicate an entire show to "songs of summer", or "songs by San Francisco bands", or "songs about food" etc. [8] Instead of focussing on a whole show, another common technique is to "put music together in sets connected by style, genres, or content, to promote continuity and help with transitions" [8]. It is jarring for the audience to listen to, e.g. a classical piece followed by a metal/hardcore song, followed by a traditional Irish jig followed by a jazz tune. Thus, many DJs group music of the same genre together in one segment, or group pieces that otherwise flow together. In our survey, many DJs also reported using the lyrics of each song to create sequences of related songs [8], e.g. if one segment is around "songs about food", then the next might be "songs about drinks". The tie between the lyrics and segments keeps listeners engaged [10].

Within each segment or even the show as a whole, the DJs we surveyed tended to order songs by pitch, tempo or loudness to manipulate the mood of the show. E.g. the DJ may start with a slow song followed by a slightly louder and faster song followed by an even louder and faster song. This will build up the energy of the show to a kind of climax at the end. Alternatively, a DJ may start with a set of fast and loud rock songs and gradually slow down towards the end of the segment or show. Of course, DJs may both increase and decrease the energy of the music within the same segment or show.

Finally, DJs are encouraged to "watch their transitions"[10]: good transitions are seamless, blending the song fading out with the following song while avoiding silent gaps between songs. E.g. a song with a metal tune that ends with a cello solo could be mixed with a classical music piece that starts with a violin, followed by a folk song that starts with a fiddle. Even though these three songs are not of the same genre, the beginning and the end of each song makes for a smooth transition, and the natural difference in pitch, loudness and tempo between each genre can allow the DJ to play with the mood of the segment. Good transitions are difficult to define; typically DJs will rely on their intuition about music to finds songs which fit best together.

While most of these practices rely on the musical knowledge and experience of DJs, some of the techniques outlined above can be formalized and used to automatically create pleasant and coherent playlists within an online or mobile music service. To our knowledge, bundling songs into coherent themes and varying the 'energy' of music within each segment have not been examined by the automatic playlist generation research.

3 Experiment Design

In order to effectively compare music scheduling strategies for music on-the-go, the following requirements must be fulfilled:

R1. Compare schedules, not music choice

The goal of the experiment is to evaluate the scheduling of music, hence the experiment design must not be biased by the choice of songs in a given playlist schedule. To ensure this, we fix the selection of music tracks used in the experiment. Each condition plays exactly the same set of songs; the sole difference is the order in which the tracks are played. The set of experiment songs must be designed to support reasonable, yet highly diverse playlists for all the strategies being compared.

R2. Evaluate music consumption *in situ*

To realistically examine the user experience of various kinds of schedules, a field experiment is necessary. The advantages of field experiments for mobile guides have been extensively discussed in Goodman et al. [15]. While the levels of potential confounding variables, such as noise levels, traffic and weather conditions cannot be kept consistent, their variation manifests itself randomly across conditions. Like [14], we consider this to be acceptable because variation in such variables is an integral part of real-world usage. By using actual locations and authentic environmental conditions, we obtain vital data on the experience of music on-the-go in practice.

R3. Be independent of users' musical tastes

To mitigate the song bias caused by users' personal preferences for certain songs or genres, users' musical preferences are measured via a Web-based survey a couple of days before the actual experiment. Users were asked to listen to and rate a superset of the songs used in the experiment on a 5-point Likert scale. The songs were presented in random order without any identifying information and include songs the users would hear eventually as part of the experiment. This rating is done on a per-song basis, but this allows us to compute an expected score for each user and schedule, depending on how much the user liked the songs in that schedule. Each user's schedule rating in the experiment is discounted by this expected score, thus removing any bias in rating caused by whether the user liked the songs in that particular playlist.

R4. Obtain clear signals of user experience

Based on prior experience, we know that people tend to rate music more positively during an experiment, which can obscure the differences among scheduling methods. That is, schedules might be rated more positively simply because users enjoyed walking on a street and listening to music. To mitigate this bias, the data analysis will focus on negative ratings, which are a clearer signal of user preferences, rather than the raw user ratings of schedules.

R5. Use experience sampling to evaluate transitions

The user experience of song transitions are a critical reflection of the quality of scheduling. To better capture the users' experience of these and reduce reliance on recall, we use the experience sampling method (ESM), asking users to rate the songs and transitions between ‘bundles’ of songs, i.e. segments of 5 songs each.

4 Experiment

We now present an experiment that examines the effect of four basic music scheduling strategies on user experience. These scheduling strategies are inspired by the practices of DJs described in Section 2.3.

4.1 Scheduling Methods

We now define the 4 kinds of scheduling methods or *schedulers* that we will evaluate in the field study. We assume that a *candidate set* of songs is already available. Given this candidate set, the task of the scheduler or scheduling method is to order these songs into a playlist such that each song is played exactly once. As an organizing strategy, each scheduler will segment the candidate set into equal-sized *bundles* (i.e. fixed subsets of songs) that all share (or differ minimally in) some feature, thus creating a smoother, less jarring listening experience.

The automatic scheduling methods we define essentially only differ in terms of which feature is used to create the bundles. Some of the methods maintain a coherent order across bundles, whereas others just order songs within bundles and then randomize the order of the bundles.

Expert Scheduling The *Expert* scheduling method is a baseline, just used for our experiments, that was created manually by an experienced DJ using the principles described in the previous section. Thus, in addition to scheduling music by pitch, tempo and genre, the schedule also takes into account song transitions and attempts to cluster songs with lyrics referring to similar entities (e.g. a food cluster may include songs with lyrics referring to sushi and pizza). This schedule can therefore be considered to be ordered both within and across bundles.

Genre Scheduling The *Genre* scheduler relies on the genre meta-data of the songs to cluster songs such that each bundle contains songs of the same genre. There is no ordering across bundles, and there is no natural order within the bundle. Genres for songs are derived from the genre meta-data specified on Wikipedia or All Music [12]. This method is an representation or instantiation of a meta-data-based scheduler. In our experiment, the genre clusters used were Jazz, Rock, Electronic, and World.

[12] http://allmusic.com

Tempo Scheduling The *Tempo* scheduler uses the amplitude or loudness of the song [13] combined with the beat of the song [14]. These features were extracted from the part of the song that was scheduled to play [15]. Given

$$S_a \equiv sort(S, amplitude), S_b \equiv sort(S, beat)$$

where $sort(S, x)$ is the playlist S ordered by feature x, and

$$idx_a \equiv idx(s_a, S_a), idx_b \equiv idx(s_b, S_b)$$

where $idx(s_x, S)$ is the rank order of feature x of song s in the ordered playlist S. For example, the song with the lowest amplitude has rank order 1 and the one with the highest amplitude has rank n, where n is the size of the candidate set to be scheduled. The tempo based order of songs $S_{a,b}$ is then defined as:

$$S_{a,b} \equiv sort(S, idx_a + idx_b).$$

In other words, the candidate set is first rank-ordered in terms of amplitude and beat separately; then re-ranked based on the sum of the two rankings. Next, the ordered list of songs, $S_{a,b}$, is split into bundles, maintaining the tempo rank order within each bundle. The resulting bundles are then ordered randomly to maintain the perception of bundles. This method is an instantiation of a content or audio-feature based scheduler.

Location Scheduling The *Location* scheduler is a novel scheduler that assumes a set of songs with associated physical locations (e.g. assigned by a DJ) and orders the songs based on the expected path taken by the listener. Thus, it always plays the song whose 'location' is closest in distance to the user. There is hence a natural order in this schedule for both within and between bundles. In the following section, we describe the song selection process for the experiment, in particular our method to pick songs for a location.

Song selection The set of experiment songs was chosen by the same experienced DJ who also then ordered them in the *Expert* schedule. The candidate set was chosen by first identifying songs for points of interest (POIs) in the experiment location, then filtering these songs to a set of 20 that would provide reasonable schedules when ordered by all the schedulers.

To choose songs for a given POI, we identified key distinguishing features of the POI and then chose audio to convey those features[16], e.g. through the melody, tempo, rhythm or lyrics of the songs. E.g., the key feature of a Thai

[13] RMS amplitude extracted using the sox tool (sox *audiofile.wav* stats | grep "RMS amplitude" | awk {'print $3'}). We got the same ordering results when extracting loudness using the RMS lev dB feature. We also found that pitch extraction did not produce any useful schedules so we dropped it.

[14] number of beats detected by the aubiocut tool (aubiocut -b -i *audiofile.wav* | wc -l)

[15] in our system we normalize this to 1 minute for all songs

[16] All the audio used in the experiment can be heard at `http://www.crowdee.com/dj`.

restaurant could be "restaurant", or "Thailand", but since there were many restaurants in the vicinity, we chose to convey the feature "Thailand" with a traditional Thai folk song. For concrete features, e.g. "coffee" (for a coffee shop) or "pizza" (for a pizza restaurant), we relied on lyrics to communicate the features, e.g., choosing "The Coffee Song" by Frank Sinatra where the lyric "they've got an awful lot of coffee in Brazil" is mentioned prominently and repeatedly in the song. For abstract features that are more difficult to convey explicitly, e.g. "India" or "France", we relied on instrumental music to remind listeners of those cultures.

The process used to filter this larger set of songs was somewhat ad hoc with trial and error to identify a candidate subset of 20 songs that yielded reasonable schedules when ordered by all four schedulers. In practice, the two main restrictions were location distribution (for the *Location* scheduler) as described in the previous section and genre coverage (for the *Genre* scheduler), i.e. we needed 5 songs in each of 4 genres. The other two schedulers (*Expert*, *Tempo*) did not impose any substantial restrictions on the chosen song set.

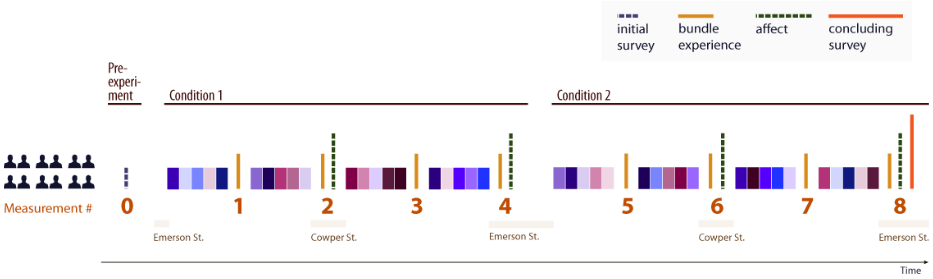


Fig. 1. Experiment design: In the pre-experiment stage, each participant rated a superset of the songs that would be used in the experiment (Measurement #0). During the field experiment, each participant participated in a guided walk during which they were exposed to 2 conditions. Each condition consisted of 4 bundles of 5 songs each, depicted by the 4 sets of 5 colored boxes. After hearing each bundle, participants were asked to report on their experience of that bundle and the transitions within that bundle. In addition, we measured their emotional response (i.e. affect of the condition) after every two bundles. At the end of the experiment (Measurement #8), participants filled out a concluding survey.

4.2 Experiment Design

Participants were asked to take a guided walk along a few blocks of a busy downtown street in Palo Alto while listening to a playlist of songs (see Figure 1). The experiment compared the user experience of four conditions, corresponding to the scheduling methods used to generate the playlist, namely: *Expert*, *Location*, *Genre* and *Tempo*. Each participant experienced 2 conditions, but the conditions are fully counterbalanced. Thus, for each condition, half of the participants experienced it as the first condition, and the other half as the second condition.

We tested the following hypotheses. We define μ as the (average) user experience for a particular condition:

Hypothesis 1:
μ is greater for the Location condition than for Expert

Hypothesis 2:
μ is greater for the Tempo condition than for Genre

The first hypothesis compares the 2 conditions that used contextual information for scheduling music. The Location schedule used knowledge of the location to order music. The Expert schedule used many different sources of information, including location and previous songs in the schedule, to order future songs. The second hypothesis examines the 2 conditions that used features of the songs themselves to order music.

4.3 Participants

We recruited 12 participants (over 90% of age 25-34 years, 4 female) from the Palo Alto area for the 45-minute experiment. The participants were only moderately familiar with the experiment location, visiting it relatively infrequently (once every month or less). Most of the participants accessed the mobile Internet daily (7/12) and about half used location-based services on a weekly or daily basis (6/12). All of them had used smartphones before, with Android and iOS tied as the most common platforms (9/12 altogether). With 12 participants, our study comprised 48 trials all together, with 6 participants per condition. We randomized the order of each pair of conditions, so that for each condition, 3 participants experienced it as the first condition and another 3 as the second condition.

Location The experiment location had to be chosen such that it could support a location-based music schedule and would be pleasant enough to walk while listening to the other schedules. We chose a busy shopping street that provided a high density of POIs and was nevertheless very ‘walkable’. The POIs chosen to be represented within the location schedule constituted a balanced mix of large and small places, as well as prominent and obscure places. They were generally equally spaced within each stretch. For the *Location* condition, the schedule was restricted to only play songs about POIs that were on the side of the street the participant was instructed to walk on. Furthermore, the songs in the *Location* schedule were ordered sequentially based on the direction of the participants’ walk to mimic a natural stroll on the street. However, the stretches corresponding to a bundle were short enough that participants could potentially walk back and forth in case they walked past a relevant POI.

4.4 Experiment Procedure

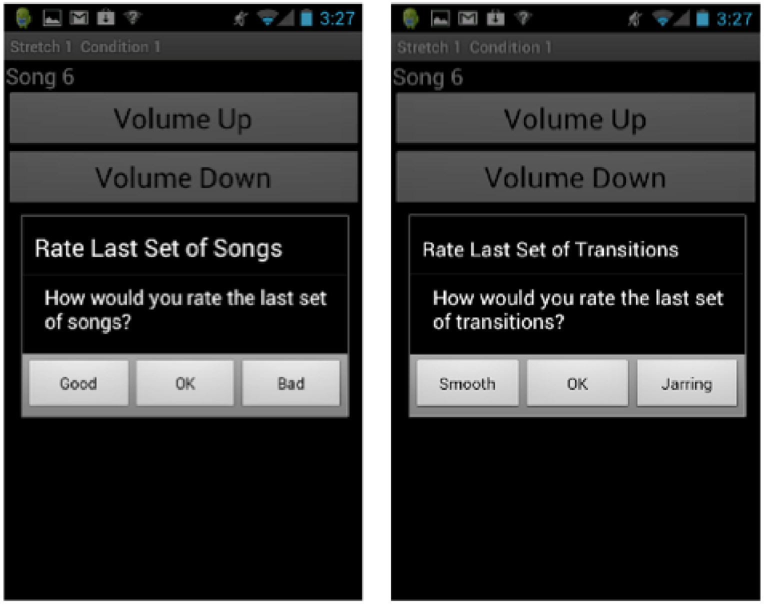
Participants were randomly assigned to two of the four conditions, which they heard while walking two loops of 4 street blocks (corresponding to the 4 bundles created for each condition). After an initial briefing on experiment procedure, the experiment device was introduced to the participants. The experiment device was an Android smartphone with a custom-built Android experiment app that presented a playlist of songs to the participant depending on the condition they were in. For the *Location* condition, the experiment app simulated the behavior of an LBS without using GPS or mobile Internet connectivity. Since the goal of the experiment was to compare the relative performance of different kinds of scheduling methods rather than evaluate the performance of a prototype, this allowed us to eliminate a potential source of confounds that might be caused by technical issues. Participants were given the prepared Android phone and a pair of earphones, and were instructed to walk at a leisurely pace.

The schedule for each condition consisted of 4 bundles of 5 songs, i.e. 20 songs played for one minute each. In line with the radio station techniques, songs were faded in and out for all the methods. The schedules were also chosen such that they were as different as possible from each other.

On the routes, the experimenter walked at a distance away from the participant to avoid disturbing and impacting the experiment, while still being able to observe and detect problems. To minimize experimenter demand effects and erroneous samples, the experimenter was available only at the end of each stretch to answer questions, while the participant filled out a mood questionnaire rating the emotional experience of that stretch. The experimenter would then also start the experiment app for the next stretch. At the end of the experiment, participants filled out a concluding survey with some open-ended questions about the experiment and their background.

As depicted in Figure 1, we took measurements four times during the experiment, i.e. after every bundle. Participants were asked to rate the bundle they just heard (on a 3-point Likert scale) in terms of basic enjoyment and smoothness of song transition (see Figure2). We obtained user ratings for both measures so that users would pay attention to both the choice and order of songs in each bundle. These two ratings are however highly correlated, so in the ensuing analysis, we treat them as an aggregate rating.

Experience Sampling After every two bundles, participants noted their emotional state, using a modified PAD (Pleasure, Arousal, Dominance) Semantic Differential Scale (PAD scale) [20, 1]. The PAD scale consists of a set of bipolar adjective pairs that are rated along a five-point scale, which corresponds to 3 dimensions of emotional response: pleasure, arousal and autonomy. To make it amenable to frequent experience sampling, we modified and condensed the scale to only include word pairs that were appropriate and easy to interpret for our experiment. This resulted in a modified scale, consisting of six pairs of words in random order.



Rate Last Set of Songs

How would you rate the last set of songs?

Rate Last Set of Transitions

How would you rate the last set of transitions?

(a) Basic bundle rating

(b) Transition rating

Fig. 2. Mobile experiment app

5 Experimental Results

5.1 Analysis

A general issue facing the analysis of measurements is making sure that the samples are independent to avoid a bias in the results, i.e. we want to extract only the true signals of user preferences. The pre-experiment survey was used to remove user bias of specific song preferences, whereas various aggregation and filter operations were used to make sure that the observations could be reasonably assumed to be independent and identically distributed (i.i.d.), a necessary condition for statistical tests. Only negative ratings were counted and the samples that were filtered out were highly correlated with the ones included, therefore the true signals they represented were already present within the data retained. If the correlations in the data set are not properly accounted for, they can have a substantial effect on the statistical conclusions.

In the following, we describe how we aggregate and filter the measured user experience for each condition. The measured experience, m, of the user for a condition is obtained for each bundle as follows:

$$m = E[m] - n_b - n_t \tag{1}$$

where $n_b \in \{0, 1\}$ denotes the number of negative basic ratings (see Figure 2) for a bundle during the walk, $n_t \in \{0, 1\}$ denotes the number of negative transition ratings, and $E[m]$ denotes the a priori expected experience based on the pre-experiment ratings. This value is obtained as follows:

$$E[m] = \lfloor n_o/2 \rfloor \tag{2}$$

where n_o corresponds to the number of negative ratings across the 5 songs in the bundle in the pre-experiment. The pre-experiment rating of both *Don't like it* and *Really dislike it* (the bottom two scores of the 5-point Likert scale) were considered negative.

In analyzing the data, we discovered that it was difficult to detect whether a user really liked a bundle or just tolerated them. Participants tended to give positive ratings as long as they did not actively dislike the bundle. This tended to obscure the differences between the different conditions. However, when users gave a negative score, it was a clear signal that the bundle resulted in a bad user experience. When we relied on the negative scores, the differences between conditions became much clearer.

We note that both the measured rating $(n_b + n_t)$ and this expected rating fall on the discrete increasing scale $\{0, 1, 2\}$. Thus $m \in \{-2, -1, 0, 1, 2\}$ is also on an increasing scale. A value of -2 for m, for instance, indicates that there were clearly more negative ratings for this bundle and condition in the experiment than one would expect from the pre-experiment ratings. We can therefore conclude that the condition had a negative impact on the experience for that user. Conversely an m-value of 2 indicates that there were clearly fewer negative ratings than one would expect from the pre-experiment ratings and thus the condition had a positive effect on the experience of that user.

5.2 User Experience

We now examine the impact of various scheduling methods on the user ratings of bundles. We take both pre-experiment and *in situ* (during the experiment walk) ratings into account. Examining the data, we found that the *in situ* ratings were generally higher than the pre-experiment ones across all the conditions, as expected. However, this does not affect the data, since we are comparing various conditions *in situ*.

The values of m are shown by bundle in Figure 3. There were 8 samples of m for each user (see Figure 1), however certain sample points were highly correlated for all users. These were samples 2, 3, 6 and 7 (corresponding to the measurement number in Figure 1), and represent the 2nd and 3rd bundles for each user in each condition. By using only the samples for the 1st and 4th bundle in each condition to compare the user experience across conditions, these large correlations disappear, and we can treat the resulting set of samples as i.i.d. The largest correlation left after this second filter is .22, which is acceptable for our purposes. With this setup we have a total of 24 independent samples of m in the experiment: 4 samples per user and 6 samples per condition. The user experience μ for a condition is then the average value of m across all 6 samples, i.e:

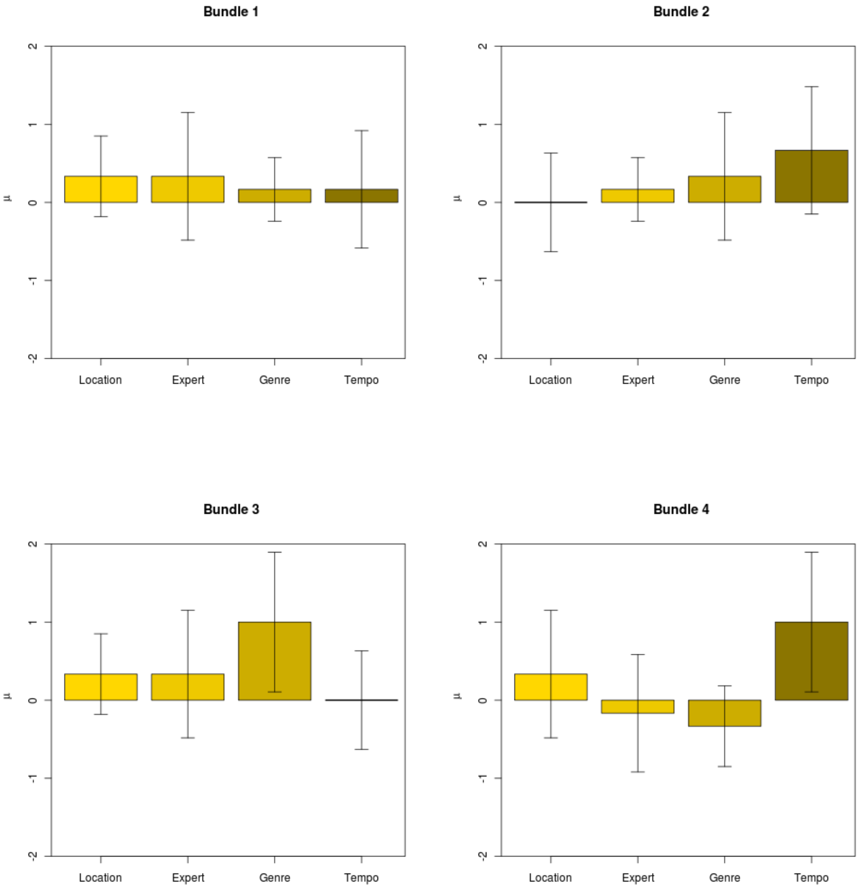


Fig. 3. Bundle-by-bundle results of user experience μ across conditions. The height of the bar represents the mean, and the error bars mark one standard deviation from the mean.

$$\mu_c = \frac{1}{6}\sum_{i=1}^{6} m_{i,c} \tag{3}$$

where c is the condition. We then conducted one-sided, two-sample, unpaired t-tests to determine whether the differences in means of the 6 samples per condition were significantly greater than 0.

Now we can evaluate our two hypotheses:

Hypothesis 1:
μ is greater for the Location condition than for Expert

Hypothesis 2:
μ is greater for the Tempo condition than for Genre

The values of μ_c and the p-values (Bonferroni-compensated) of the corresponding t-tests are shown in Table 1.

Table 1. Results of Bonferroni-compensated t-tests

Hypothesis			p-value
	μ_{Loc}	μ_{Exp}	$H_0 : \mu_{Loc} \leq \mu_{Exp}$
1	.33	.08	.40
	μ_{Tem}	μ_{Gen}	$H_0 : \mu_{Tem} \leq \mu_{Gen}$
2	.58	$-.08$	.04

We recall that a higher mean of m signifies a better experience of that condition. We note that the *Location* condition appears better than the *Expert* condition but not at a significant level. However, the *Tempo* condition is significantly better than the *Genre* condition.

Conclusion 1:
We do not find significant support for Hypothesis 1, that the Location condition yields a better user experience than the Expert condition
Conclusion 2:
We find significant support for Hypothesis 2, that the Tempo condition results in a better experience than the Genre condition

6 Discussion

In the concluding survey, most participants (8/12) reported being generally satisfied with the quality of scheduling within the conditions. The experience on the street was more enjoyable than that of the online pre-experiment. The different scheduling strategies do seem to affect the user experience, indicating that our experimental manipulation worked. Our study participants preferred the *Tempo* and *Location* conditions and were less favorable to the *Expert* and *Genre* conditions. One possible reason why the *Expert* condition did not perform as well might be that the candidate set was too small and restricted (by location and genre distribution) in our experiment to allow for the more sophisticated human techniques to fully play out.

The strongest result from our study is that *Tempo* (a linear combination of amplitude and beat rankings) yields a better user experience than *Genre* bundling. This could be because a meta-data based approach like genre scheduling requires more cultural knowledge and music expertise to appreciate whereas the *Tempo* condition is more neutral and can work well across all subjects. The genre scheduling performed particularly poorly in the last bundle that played World music. World music is itself a very heterogeneous category, but its poor performance highlights the problem of relying on, to some level, subjective categorizations. Another possible reason that *Tempo* performed well compared to *Genre* could be that it made people aware of the pace of the walk and matched it within certain bundles. A couple of people reported in the concluding survey that this was very enjoyable and suggested that the system should automatically match the tempo of the songs to the pace at which the user was walking[17].

Our experiment design enabled us to study the various scheduling strategies in a field experiment and extracting statistically significant results. We used the smallest possible number of subjects to be able to derive basic statistical test results on our key conditions. The scheduling strategies we used were relatively simple. More complex strategies, using for example, features from music intelligence platforms like the Echo Nest, could be successfully evaluated for mobile consumption using our experiment design. Although this is not common practice, using purely negative ratings to obtain clearer user signals is likely to benefit the analysis of other user experience studies. Our filtering strategy to remove highly correlated samples and clean user signals could be fruitfully used by other small user experience studies to obtain stronger results.

7 Conclusions

In this work, we evaluated different scheduling methods on the user experience of music consumed on the go. We show that different schedules do affect the user experience and that techniques developed by radio station DJs to produce cohesive and smooth radio shows can be applied successfully for scheduling mobile music. Bundling, in particular, seems to be a useful technique that can help organize disparate content into coherent, aurally pleasant clusters. However, simply replicating rules as used by radio station DJs may not always yield the best experience, given that location relevance appeared to be an important determinant of the mobile music user experience. We found that ranking the songs within the set of candidate songs with respect to the amplitude and beats and then ordering the songs using a linear combination of these two rankings yielded the best user experience.

We presented the results of a survey of DJs for college radio stations, with the heuristics and techniques they use to create playlists for radio shows and how they ensure smooth and engaging user experiences. We also presented a field experiment design to evaluate scheduling strategies in terms of user experience. Our work has implications for music recommendation systems in improving the

[17] In fact, there are several applications that do this already, e.g. SynchStep (`synchstep.com`) and TrailMix (`trailmixapp.com`).

user experience of music consumed on the go. We consider this to be a promising research approach and encourage future work in this area.

Acknowledgments. This work was completed during the authors' time at Hewlett Packard Labs. The authors would like to thank Bernardo Huberman, senior HP fellow, for his guidance.

References

1. Agarwal, A., Meyer, A.: Beyond usability: Evaluating emotional response as an integral part of the user experience. In: Proceedings of CHI 2009. pp. 2919–2930. CHI '09, ACM (2009)
2. Alghoniemy, M., Tewfik, A.H.: User-defined music sequence retrieval. In: Proceedings of the eighth ACM international conference on Multimedia. pp. 356–358. MULTIMEDIA '00, ACM, New York, NY, USA (2000)
3. Ankolekar, A., Sandholm, T., Yu, L.: Play it by ear: A case for serendipitous discovery of places with musicons. In: Proceedings of CHI 2013. pp. 2959–2968. CHI '13 (2013)
4. Avesani, P., Massa, P., Nori, M., Susi, A.: Collaborative radio community. In: Proceedings of the Second International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems. pp. 462–465. AH '02, Springer-Verlag, London, UK, UK (2002)
5. Bosteels, K., Pampalk, E., Kerre, E.E.: Evaluating and analysing dynamic playlist generation heuristics using radio logs and fuzzy set theory. In: Proceedings of ISMIR (2009)
6. Bull, M.: Sounding out the city : personal stereos and the management of everyday life. Berg, Oxford : (2000)
7. Bull, M.: Sound moves: iPod culture and urban experience. Routledge (2008)
8. CFRC 101.9 FM: CFRC Volunteer Manual (Dec 2010), http://cfrc.ca/blog/wp-content/uploads/2009/03/volunteer-manual-32.pdf
9. CFUV 101.9 FM: CFUV Orientation Guide (Nov 2009), http://cfuv.uvic.ca/cms/wp-content/uploads/2012/03/Orientation-Manual-09-10-6th-edition-1.pdf
10. CJSR 88.5 FM: Music Show Basics (Dec 2001), http://www.firststage.ca/csirp/training/articles /musicshowbasics.html
11. Denora, T.: Music in everyday life. Cambridge University Press (2000)
12. Eastman, S., Ferguson, D.: Media programming: strategies and practices. Thomson/Wadsworth (2008)
13. Gaye, L., Holmquist, L.E., Behrendt, F., Tanaka, A.: Mobile music technology: report on an emerging community. In: Proceedings of the 2006 conference on New interfaces for musical expression. pp. 22–25. NIME '06 (2006)
14. Goodman, J., Brewster, S.A., Gray, P.: How can we best use landmarks to support older people in navigation? Journal of Behaviour and Information Technology 24, 3–20 (2005)
15. Goodman, J., Brewster, S., Gray, P.: Using field experiments to evaluate mobile guides. In: Proc. of HCI in Mobile Guides, workshop at Mobile HCI 2004 (2004)
16. Hauver, D., French, J.: Flycasting: using collaborative filtering to generate a playlist for online radio. Web Delivering of Music, 2001. Proceedings. First International Conference on pp. 123–130 (23-24 Nov 2001)

17. Hayes, C., Cunningham, P.: Smart radio: Building music radio on the fly. In: Expert Systems 2000. pp. 2–6. ACM Press (2000)
18. Komulainen, S., Karukka, M., Häkkilä, J.: Social music services in teenage life: A case study. In: Proceedings of the 22Nd Conference of the Computer-Human Interaction Special Interest Group of Australia on Computer-Human Interaction. pp. 364–367. OZCHI '10 (2010)
19. Logan, B.: Content-based playlist generation: Exploratory experiments. In: Proceedings of 3rd International Conference on Music Information Retrieval. Paris, France (2002)
20. Mehrabian, A., Russell, J.A.: An approach to environmental psychology. M.I.T. Press, Cambridge, MA, USA (1974)
21. Nemirovsky, P., Davenport, G.: Guideshoes: navigation based on musical patterns. In: CHI '99 extended abstracts on Human factors in computing systems. pp. 266–267. CHI EA '99 (1999)
22. Nettamo, E., Nirhamo, M., Häkkilä, J.: A cross-cultural study of mobile music - retrieval, management and consumptiom. In: OZCHI '06. pp. 87–94. ACM (2006)
23. Pampalk, E., Pohle, T., Widmer, G.: Dynamic playlist generation based on skipping behavior. In: Proceedings of ISMIR (2005)
24. Pauws, S., Eggen, B.: Pats: Realization and user evaluation of an automatic playlist generator. In: In ISMIR. pp. 222–230 (2002)
25. Pauws, S., Verhaegh, W., Vossen, M.: Music playlist generation by adapted simulated annealing. Inf. Sci. 178(3), 647–662 (Feb 2008)
26. Ragno, R., Burges, C.J.C., Herley, C.: Inferring similarity between music objects with application to playlist generation. In: Proceedings of the 7th ACM SIGMM international workshop on Multimedia information retrieval. pp. 73–80. MIR '05, ACM, New York, NY, USA (2005)
27. Surhone, L., Tennoe, M., Henssonow, S.: Radio Computing Services
28. Warren, N., Jones, M., Jones, S., Bainbridge, D.: Navigation via continuously adapted music. In: CHI EA '05. pp. 1849–1852 (2005)