Adaptive Radio: Achieving Consensus Using Negative Preferences

Dennis L. Chao **Justin Balthrop** **Stephanie Forrest**
Department of Computer Science
University of New Mexico
Albuquerque, NM 87131 U.S.A.
{dlchao,judd,forrest}@cs.unm.edu

ABSTRACT

We introduce the use of *negative preferences* to produce solutions that are acceptable to a group of users. Using negative preference profiling, a system determines which solutions are unsatisfactory to individual users, and it is assumed that the remaining solutions are satisfactory. To satisfy all members of the group, the system can propose solutions that are not unsatisfactory to any of the group's members. This approach can find a large set of solutions that are acceptable to a group and simplify user profiling. To demonstrate these benefits, we implemented Adaptive Radio, a system that selects music to play in a shared environment. Rather than attempting to play the songs that users *want* to hear, the system avoids playing songs that they do *not* want to hear. Negative preferences can potentially be applied to other domains, such as information filtering, intelligent environments, and collaborative design.

INTRODUCTION

It can be difficult for groups of people to agree on anything. These decisions can be as mundane as deciding where your office wants to go for lunch or what movie you and your friends want to see, or they can be as important as finding a single solution that satisfies a group of collaborating designers. We have previously proposed the use of negative preferences as a solution to this problem [3], and in this paper we present a new application, Adaptive Radio, to illustrate the approach, which can easily be applied to other domains, such as information filtering, intelligent environments, and collaborative design.

Adaptive Radio is a system that broadcasts songs to a group of listeners who share an environment. Most music systems try to determine what kinds of songs users prefer, usually using surveys or online profiling of their listening habits. In contrast, Adaptive Radio keeps track of the songs that each of the users does *not* like and avoids playing songs disliked by any of the current listeners. We believe that these negative preferences are easier to collect and manipulate than positive ones in a background music system. Using negative preferences allows Adaptive Radio to play songs that are unfamiliar or simply acceptable to users in addition to their favorites. This not only increases the variety of music that a single user will hear, but it also allows a group of people to listen to music that they can all enjoy. If only positive preferences were used (i.e., songs that the users explicitly request), the preferences of a group of users would likely have little or no overlap, and no music could be played without disregarding at least some of the users' preferences. Using negative preferences, it is more likely that everyone's preferences can be accommodated, and we call the solutions that satisfy everyone *consensus solutions*. Negative preferences are also more natural to register while using the application—users only need to interact with Adaptive Radio when the music is not desirable. When the user is satisfied with the music, no action needs to be taken.

In this paper, we start with an overview of recommender systems and software that makes recommendations to groups. This section is followed by a description of negative preferences and their application to making recommendations to groups. Next, we describe the Adaptive Radio application and our experiences with the system. Finally, we discuss the challenges of playing music to groups in light of our findings.

RELATED WORK

Our goals in implementing Adaptive Radio are distinct from those of collaborative filtering. Collaborative filtering uses the preferences of others to help an individual make choices [8,14,21]. A typical example is a system that recommends items to purchase based on individuals with a similar purchase history. By harnessing the collective preferences of many individuals, such systems can infer similarity between items without needing to understand the relationship between them. This approach is useful when it is difficult for a program to quantify similarities between items, such as for art or music. Unlike collaborative filtering, the negative preferences approach used by Adaptive Radio is simply a way for groups of users to consolidate their preferences so that group recommendations can be made. However, Adaptive Radio could potentially incorporate collaborative filtering techniques to determine the similarity between songs.

A few other systems recommend items to groups instead of individuals. MusicFX [15] selects music stations that are broadcast to a gym full of people. The members of the gym must rate all the stations beforehand, and MusicFX plays one of the stations with the highest average rating. The system thus attempts to maximize the happiness of the group. One of MusicFX's shortcomings is that it would be difficult for it to scale to a large number of musical choices because the users need to be able to rate all of the stations for MusicFX to produce good recommendations. Flycast [7] uses a nearly identical method to generate the playlist of an online radio station. GroupCast [16], developed by the same research group as MusicFX, is a conceptually similar system that selects content for a public display system. Unfortunately, the researchers found that the necessary user profiles would have been too large for any user with a reasonable amount of patience to complete. In addition, without extensive profiles it was difficult to find appropriate intersections of user preferences to put on the GroupCast displays. Instead, it displayed content that was interesting to *one* of the users, hoping that by chance others would have similar interests. Flytrap [4] addresses the profile-building problem by unobtrusively monitoring each user's personal MP3 player to determine the user's musical preferences. Like Adaptive Radio, it determines what music to broadcast to groups based on the members' profiles, but the method it uses to combine these preferences, described below, is different.

PolyLens [20] recommends movies to small groups of people who watch movies together. It applies a standard collaborative filtering algorithm to find recommendations for each of the group members, and then it combines the results to make a group recommendation. Unlike MusicFX, PolyLens attempts to satisfy all users to some degree, without necessarily maximizing the group average. PolyLens bases its recommendations on the expected happiness of the *least* satisfied group member. Therefore, a movie that is barely acceptable to each of the group members is recommended over one that one person would hate but everyone else would enjoy immensely.

These group recommendation systems illustrate two voting-based approaches to combining the preferences of several individuals. MusicFX and Flycast attempt to please the majority at the expense of a dissenting minority. PolyLens and Adaptive Radio make concessions to the minority opinion to ensure that the recommendation accommodates everyone. The schemes used by PolyLens and Adaptive Radio resemble *approval voting*, a voting procedure that is more fair than procedures that require ranking of preferences [27].

An alternative to voting is to average the preferences of the users, an especially tempting option when the tastes of the users do not intersect. Unfortunately, this can lead to unexpected and undesirable outcomes, as illustrated by the "Most Wanted Song" project [23]. To create America's "most wanted" song, a group of artists used a web-based poll to determine the attributes of a desirable song, such as tempo, length, and instrumentation. They tallied the results of over 500 responses to create a song that most people should like: a curiously bland five-minute R&B love ballad. The artists also recorded the song that most people should dislike: a painful thirty-minute track of an opera singer rapping to tuba and accordion accompaniment with a gratuitous bagpipe solo and shrill children's chorus added to repel as many listeners as possible. Neither song is particularly pleasant to hear. Clearly it can be difficult, and perhaps undesirable, to combine the preferences of many individuals. Unfortunately this approach has been taken by at least one group recommender system; Flytrap [4] reconciles disparate musical tastes by finding genres that are "between" them.

We have developed a new approach, the *information immune system*, to reconcile multiple user preferences [2]. Using this approach, a system remembers what users do *not* want instead of remembering what they *do* want. These "negative" preferences can be combined easily. Rather than finding intersections or averages of user preferences, one can simply take the union of all users' negative preferences and assume that the choices that are outside this set are likely to be acceptable to the group. The advantages of using this technique for Adaptive Radio are outlined in the next section. Although we do not outline the parallels between this approach and the immune system in this paper, those interested in the biological motivation behind it can refer to [3].

NEGATIVE PREFERENCES

A system that uses negative preferences acts like a filter that blocks items that the user does not want and allows everything else to pass through. Such a system can initially assume that all solutions are acceptable and the user must notify it when an undesirable solution is encountered. In the future, the system will censor this undesirable solution for the user. It can gradually learn a user's preferences as he or she provides negative feedback over time. Eventually, the information filter will allow only desirable items to reach the user. Users do not need prior knowledge of the solutions—they simply need to know what they don't like. In contrast, a positive preference scheme that keeps track of what the user *wants* usually requires that user preferences be determined beforehand. User profiling is tedious, often requiring the use of surveys, and the resulting profiles are often incomplete due to the impatience or forgetfulness of the users. Negative preference applications do not require explicit user profiling processes—the users can train it by expressing dissatisfaction with its output.

A negative preference system would not be practical if a user needed to reject every undesirable item explicitly. An essential component is a distance or similarity measure between data items. When a user rejects a candidate, it can be assumed that the user would dislike similar candidates as well. Thus, rejecting one data item effectively censors a set of similar ones. This similarity metric might be difficult to implement in practice, especially when the domain is as subjective as music. Other schemes, such as pattern classification, could also be used to generalize a user's preferences based on a few exemplars.

A major benefit of using negative preferences is the ease with which user preferences of multiple individuals can be combined. One simply takes the union of the negative preferences of the individuals to find the group's preferences. This combined preference set will filter out items that are disliked by any of the group members. We call the remaining solutions *consensus solutions* because they have been implicitly approved by everyone. Although this is formally equivalent to finding the intersection of the positive preferences of the same group, it can be more effective in practice.

Because knowledge of user preferences, positive or negative, is typically incomplete, a positive preferences scheme is likely to underestimate the number of solutions a user would tolerate while the negative scheme would overestimate this value. Because the intersections of multiple users' preferences can be hard to find, if they exist at all (a problem encountered by GroupCast developers [16]), it is preferable to err on the side of overestimating rather than underestimating the space of acceptable solutions.

Negative preference systems might enhance the influence of group members who hold minority opinions. In group decision making, social influence can pressure people to change their expressed views. These influences can be normative (the desire to conform) or informational (learning from others to inform one's own judgement) [5]. The negative preference approach can reduce the normative processes that can suppress minority opinions. Firstly, it requires that the group reaches consensus, which might alleviate the need for the minority to accept the majority opinion [1]. This would not be the case if a majority voting scheme were used. Secondly, the decision-making process can be made anonymous, which further reduces the social pressures exerted by the majority. Negative preference systems implicitly favor informational influences, which can increase the influence of the minority [18, 26]. Experienced group members with the largest negative preference profiles exert the most control over the group decision by censoring more of the decision space. Therefore, knowledgeable members holding a minority opinion have the opportunity to convert members of the majority to their side. However, the effects of social influence are not straightforward in negative preference systems. If someone censors a solution, the other members will not be exposed to it and are given no opportunity to censor it also. One influences the group's decisions but not the other members' profiles. If positive preferences were used, then members could explicitly agree with the group's choices and make their profiles conform to them, leading to uniformity among preference profiles.

We believe that the use of negative preferences can be extended to other domains [3]. Tasks that require catering to the desires of multiple individuals could benefit from this approach. For example, intelligent environments that dynamically change the contents of displays based on who is observing them could use negative preferences. Negative preference systems can also be used as an aid to creativity. A designer could brainstorm by viewing random design candidates filtered using his or her set of negative preferences. Initially, a large number of unacceptable candidates would be presented, but as the designer trains the system it would present a wide range of acceptable candidates. Collaborating designers could use the same technique by combining their sets of negative preferences. Only solutions that are acceptable to all would be likely to survive the filtering by the designers' preferences. Groups appear to be less effective at generating ideas than separate individuals [24]. Using a negative selection scheme removes the need for collaborators to interact, which might lead to more productive brainstorming.

Figure 1. The Adaptive Radio user interface.

ADAPTIVE RADIO

The Application

Adaptive Radio is a music server that broadcasts to a group of people.[1] It chooses its song selections from MP3s contributed by users of the system. Adaptive Radio uses Icecast [11] to stream music to users' personal computers, but users must log on if they want their preferences to influence the choice of songs. It has a simple web-based interface (Figure 1). After logging on, the user can indicate that he or she does not like the currently playing song by pushing the "censor" button, which causes the Adaptive Radio server to remember not to play this song or similar ones in this user's presence again. Pressing the "skip" button causes the server to stop playing the current song and to randomly choose another. Having separate censor and skip functions allows users to register dislike for a song without interrupting the flow of music or to quickly survey the musical choices (channel-surf) without registering dislike.

Adaptive Radio constructs musical preference profiles for each person using only the list of songs that are disliked by the users, implicitly assuming that unrated songs are acceptable until proven otherwise. If instead one played songs that users already know that they like, unfamiliar music and music that has recently been added to the system would be

[1] Adaptive Radio's source code has been released under the GNU General Public License (GPL) and is available at http://www.cs.unm.edu/~dlchao/radio/index.html.

unrated, and therefore unused by the system. By including unfamiliar music and songs that are not preferred but still acceptable to users, Adaptive Radio can select from a much larger set of songs.

Adaptive Radio avoids playing songs similar to those that have been rejected by any of the users who are currently listening, resulting in a song playlist that should please all users. If there is only one listener, the system will play music that this person likes. As more people arrive, the selection of music will narrow to accommodate the listening preferences of the new users. In effect, all users can veto song selections. Although this interface can be frustratingly spartan for music aficionados, we believe that its simplicity allows us to observe the effects of using negative preferences most clearly. A music selection could be constructed to use both positive and negative preferences if desired.

Because it is difficult to determine automatically the similarity between songs, Adaptive Radio assumes that only songs on the same album are similar to each other. Therefore, if a user rejects a song, the rest of the album is censored as well. This crude similarity measure seems to work well in practice, but an area of future research is the adoption of collaborative filtering techniques to create a more accurate metric.

Evaluation and Discussion

We evaluated Adaptive Radio by making informal observations of its use, examining users' Adaptive Radio music profiles, and giving users a short survey a few months after it was installed in our office. The survey results indicate that users were happy with the performance of Adaptive Radio. Prior to its installation, music was rarely played in the office. This was due in part to the fear of bothering coworkers with one's own musical selections. The participation of all office members in the musical selection alleviated this concern. Users quickly became comfortable with the user interface, which allows them to reject songs with little conscious effort. Registering disapproval became a nearly automatic reaction to undesirable music, as evidenced by the channel-surfing behavior during which a user would quickly reject several consecutive songs without interrupting his or her work. People seem to find it more natural to reject songs than to provide positive feedback to a music selection system. When Adaptive Radio is playing desirable music, the listener should not need to think about the system. When undesirable music intrudes upon a listener's consciousness, he or she can quickly register disapproval.

According to our survey, Adaptive Radio introduced some users to music with which they were not previously familiar but now appreciate. Some users who had seemingly different musical tastes discovered that they enjoyed the music of their coworkers. These serendipitous newfound musical preferences would be difficult to discover using a positive preference approach like MusicFX's that preferentially plays what the listeners already know they like. Other users with little obvious overlap in musical tastes have noticed that Adaptive Radio often plays Simon and Garfunkel songs when they are in the room together. We soon realized that fast or loud songs are prone to rejection by people trying to work, which was confirmed in our survey results.

We examined the Adaptive Radio profiles of the four most regular users. Each of these users explicitly rejected between 45 and 117 songs, and their combined profiles censored 1498 out of 1862 songs in the system, or 80%. Most of the albums that were not rejected by anyone were folk (including Simon and Garfunkel) and jazz, while albums from louder and faster genres like funk and electronic dance music were consistently rejected. The survey indicated that people generally agreed on what they did not want to listen to at work (usually characterized as fast or hard music), but they did not agree on the types of music they would like to hear. Therefore, a positive preferences scheme that queries users for their musical preferences would have had found it difficult to find an intersection of all the individual user preferences.

The songs that are least likely to be rejected are slow, quiet, and familiar, in other words elevator music. Although the term "elevator music" is usually used pejoratively, Paul Simon does not object to this characterization of his music, claiming that "it's nice to have any song that you write played in an elevator" [22]. Our *passive* musical preferences can be quite different from our *active* ones. While we might enjoy dynamic and challenging music in a concert setting, at work we might prefer something less intrusive. Background music that calls attention to itself could be distracting. In a workplace with broadcast music, *everyone* must be accommodated, even if compromising seems unsatisfactory to the majority. A Muzak executive describes what can happen when employees try to choose their own music:

> In an office for a garment factory outside of Atlanta, the workers got tired of the Muzak and used a radio for their background music. If they turned on rock, 25 percent of the people in the workplace didn't like it. So they got a committee together and took a vote. They played the classical station, and only 10 percent of the people ended up liking it. So they tried a country station, and 60 percent didn't like it. They had another meeting. They decided on one day for each format: country one day, classical the next, disco for maybe half a day. But the 10 percent who liked whatever was playing got tired of people glaring at them. Finally the office manager called us and asked if they could have the Muzak back. It proved what I was doing was working. Muzak proved the least of all possible evils. [13]

The advantage that Adaptive Radio has over the often-ridiculed fare of the Muzak corporation is that it can cater specifically to the occupants of a room. The users are likely to appreciate the fact that they have control over the music being broadcast [15]. If they happen to share musical tastes, the variety of acceptable music can be large; if not, the range is likely to be small and possibly less than satisfactory. As one Muzak programmer explains, "There are literally 90 million people listening to Muzak per day. It's a real challenge to put something together that's going to

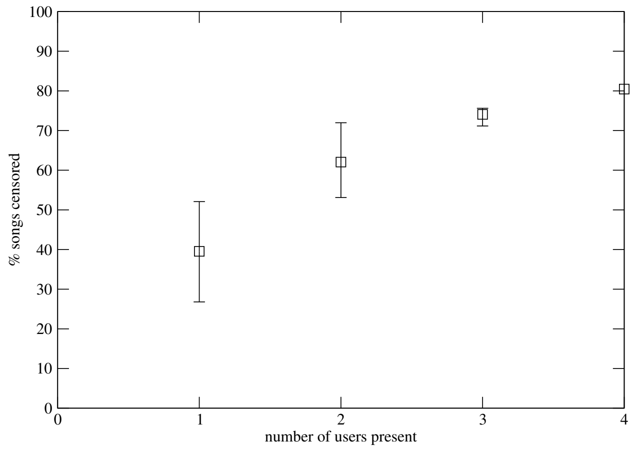

Figure 2: The relationship between the number of users and the percentage of songs Adaptive Radio must censor. The boxes represent the mean percentage of songs that Adaptive Radio censors in the presence of a given number of users, and the bars indicate the minimum and maximum values.

please everyone… Since we have so many people listening at once, we are forced to amalgamate" [13]. As the "Most wanted song" project demonstrated, amalgamation might be a bad idea, hence the nearly universal dislike of Muzak [19].

We explored the relationship between the number of users present and the number of songs that Adaptive Radio could play to them. As mentioned earlier, when the four most regular users were present, the system would censor 80% of the songs. When only one of these users was present, it would censor between 27% and 52% of songs, depending on the user. For two users, we computed the percentage of songs Adaptive Radio would censor for all possible pairs of these four users and averaged the results, and for three users we did the same for all possible subsets of three users. The results are shown in Figure 2. The percentage of songs censored rises rapidly with the number of users. For one and two users, the number of songs censored is heavily dependent on the particular users involved, as indicated by the large distance between the minimum and maximum values in Figure 2. However, with three users, the minimum and maximum percentages of songs censored converges, indicating that no matter which three users are present, the song selection will be similar. Adding a fourth user does not greatly increase the number of songs censored. In other words, the presence of only three users in our office guarantees a narrow song selection for Adaptive Radio. Unfortunately, this implies that once three people are in the office, music tends toward Muzak.

We believe that reconciling the musical tastes of even a modest number of random individuals (perhaps as few as three!) can result in "lowest common denominator" music. This phenomenon has been observed in broadcast radio. As radio station playlists were generated less by individual DJs and more by market research, the playlists got smaller and catered to the lowest common denominator while alienating listeners with unusual tastes [17]. However, we believe that in many situations, diverse people can enjoy a wide variety of music together. The success of store-branded music CDs supports this belief. Many retailers order custom-made compact discs featuring musical selections that reinforce their store's image [6]. The shoppers are not united by their musical preferences but by the goods and image offered by a particular store. The creators of these music compilations are free to cross genre boundaries if they are in keeping with the store's image, and many of these albums have sold very well. Given the diversity of songs on many of these compilations, it appears that in the right environment groups of people can listen to a broad range of music together. We believe Adaptive Radio would also be effective in the home, where the number of occupants is small and the range of music that can be played might be larger than that in an office.

Adaptive Radio could also be used to create non-intuitive groupings of individuals who share musical tastes, which would be useful when a limited number of broadcast channels needs to accommodate a large number of listeners. Normally, radio stations specialize in genres of music, and listeners must choose among them based on these predetermined categories of music. If Adaptive Radio were to partition the listeners automatically based on their preferences and cater to each group's collective preferences, it could generate novel playlists that cross established genre boundaries. For example, one could use a greedy algorithm to assign users to groups in such a way that maximizes the overlap of musical dislikes within the groups. Alternatively, a more sophisticated algorithm for clustering users, such as the one proposed in [28], can be adapted to use negative preferences. The broadcaster could then choose music for each group that would reflect the preferences that the listeners in each have in common.

CONCLUSIONS

We have developed a system, Adaptive Radio, that plays music to a group of people by learning what the users do not want to hear and avoiding these songs. It is a simple program that illustrates the use of negative preferences to cater to a group of individuals. Using negative preferences expanded the set of songs that could be played and provided a natural way to combine user preferences. By requiring feedback only when the user is dissatisfied with its performance, using Adaptive Radio does not distract users from their work. Our results suggest that the system will play bland music in the workplace once there are three or more people present, but this may be the inevitable result of consensus and of the nature of the office environment. We believe that in environments with fewer people, such as the home, or with people who have other interests in common, such as shoppers in a particular store, Adaptive Radio could be an effective mediator of differing musical tastes.

The negative preference technique can easily be applied to other tasks, such as information filtering, intelligent environments, and collaborative design. Group-mediated solutions will become essential as computers migrate from our desktops to our daily lives in the form of intelligent environments and ubiquitous computing applications. Most research in these areas emphasizes the preferences of a single individual [9, 10, 12, 25], but for many environments it will be important to satisfy multiple occupants simultaneously. Without an intelligent means of reconciling individual preferences, these personalized technologies are likely to follow the path of the Walkman (personalized but antisocial service) or Muzak (impersonal and catering to the lowest common denominator).

Acknowledgments
We thank Allan MacLean and Bob Axelrod for providing helpful feedback on this manuscript. The authors gratefully acknowledge the support of the National Science Foundation (ANIR-9986555), the Office of Naval Research (N00014-99-1-0417), the Defense Advanced Research Projects Agency (AGR F30602-00-2-0584), the Intel Corporation, and the Santa Fe Institute.

REFERENCES

1. B. Beersma and C. K. W. De Dreu. Integrative and distributive negotiation in small groups: Effects of task structure, decision rule, and social motive. *Organizational Behavior and Human Decision Processes*, 87:227–252, 2002.
2. D. L. Chao and S. Forrest. Generating biomorphs with an aesthetic immune system. In R. Standish, M. A. Bedau, and H. A. Abbass, editors, *Artificial Life VIII: Proceedings of the Eighth International Conference on the Simulation and Synthesis of Living Systems*, pages 89–92, Cambridge, Massachusetts, 2003. MIT Press.
3. D. L. Chao and S. Forrest. Information immune systems. *Genetic Programming and Evolvable Machines*, 4(4):311–331, 2003.
4. A. Crossen, J. Budzik, and K. J. Hammond. Flytrap: Intelligent group music recommendation. In *Proceedings of the 7th International Conference on Intelligent User Interfaces*, pages 184–185, New York, 2002. ACM Press.
5. M. Deutsch and H. B. Gerard. A study of normative and informational social influence upon social judgment. *Journal of Abnormal and Social Psychology*, 51:629–636, 1965.
6. J. Flaherty. Music to a retailer's ears. *The New York Times*, 4 Jul:C1, 2001.
7. J. C. French and D. B. Hauver. Flycasting: On the fly broadcasting. In *Proceedings of the Second DELOS Network of Excellence Workshop on Personalization and Recommender Systems in Digital Libraries*, pages 89–93, 2001.
8. D. Goldberg, D. Nichols, B. M. Oki, and D. Terry. Using collaborative filtering to weave an information tapestry. *Communications of the ACM*, 35(12):61–70, 1992.
9. S. R. Hedberg. After desktop computing: A progress report on smart environments research. *IEEE Intelligent Systems*, 15(5):7–9, 2000.
10. *House_n Living Laboratory*. School of Architecture and Planning, Massachusetts Institute of Technology, Cambridge, Massachusetts, 2001.
11. Icecast. http://www.icecast.org, 2003.
12. C. D. Kidd, R. J. Orr, G. D. Abowd, C. G. Atkeson, I. A. Essa, B. MacIntyre, E. Mynatt, T. E. Starner, and W. Newstetter. The aware home: A living laboratory for ubiquitous computing research. In N. A. Streitz, J. Siegel, V. Hartkopf, and S. Konomi, editors, *Proceedings of the Second International Workshop on Cooperative Buildings (CoBuild'99)*, volume 1670 of *Lecture Notes in Computer Science*, pages 191–198, Berlin, 1999. Springer.
13. J. Lanza. *Elevator Music: A surreal history of Muzak, easy-listening, and other moodsong*. St. Martin's Press, New York, 1994.
14. T. W. Malone, K. R. Grant, F. A. Turbak, S. A. Brobst, and M. D. Cohen. Intelligent information sharing systems. *Communications of the ACM*, 30(5):390–402, 1987.
15. J. F. McCarthy and T. D. Anagnost. MusicFX: An arbiter of group preferences for computer supported collaborative workouts. In *Proceedings of the ACM 1998 Conference on Computer Supported Cooperative Work*, pages 363–372, New York, 1998. ACM Press.
16. J. F. McCarthy, T. J. Costa, and E. S. Liongosari. UniCast, OutCast & GroupCast: An exploration of new interaction paradigms for ubiquitous, peripheral displays. In *Workshop on Distributed and Disappearing User Interfaces in Ubiquitous Computing at the SIGCHI Conference on Human Factors in Computer Systems (CHI 2001)*, New York, 2001. ACM Press.
17. L. McShane. Radio, radio: Where did all the music go? *The Associated Press*, 2002.
18. S. Moscovici. Towards a theory of conversion behavior. In L. Berkowitz, editor, *Advances in experimental social psychology*, pages 209–239. Academic Press, London, 1980.
19. No thank you for the Muzak. *The Guardian Society Pages (London)*, 1 Dec:2, 1999.
20. M. O'Connor, D. Cosley, J. A. Konstan, and J. Riedl. PolyLens: A recommender system for groups of users. In W. Prinz, editor, *Proceedings of the Seventh European Conference on Computer Supported Cooperative Work*, pages 199–218, New York, 2001. Kluwer Academic.
21. P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl. GroupLens: An open architecture for collaborative filtering of Netnews. In R. K. Furuta and C. M. Neuwirth, editors, *Proceedings of the ACM 1994*

Conference on Computer Supported Cooperative Work, pages 175–186, New York, 1994. ACM Press.

22. T. Schwartz. Interview: Paul Simon. *Playboy*, 31(2):49–51,163–74, 1984.

23. D. Soldier, V. Komar, and A. Melamid. *The People's Choice Music*. Mulatta Records, New York, 2002.

24. W. Stroebe and M. Diehl. Why groups are less effective than their members: On productivity loss in idea generating groups. *European Review of Social Psychology*, 5:271–304, 1994.

25. The user in control. *Philips Research Password*, 3:10–13, 2000.

26. A. van Hiel and V. Franssen. Information acquisition bias during the preparation of group discussion: A comparison of prospective minority and majority members. *Small group research*, 34(5):557–574, 2003.

27. R. J. Weber. Approval voting. *J. Econ Perspectives*, 9(1):39–49, 1995.

28. T. Wong, R. H. Katz, and S. McCanne. An evaluation of preference clustering in large-scale multicast applications. In *Proceedings of IEEE INFOCOM*, pages 451–460, 2000.