Cobi: A Community-Informed Conference Scheduling Tool

Juho Kim, Haoqi Zhang, Paul André, Lydia B. Chilton, Wendy E. Mackay, Michel Beaudouin-Lafon, Robert C. Miller, Steven P. Dow

Cobi: A Community-Informed Conference Scheduling Tool

Juho Kim[1] **Haoqi Zhang**[1,2] **Paul André**[3] **Lydia B. Chilton**[4] **Wendy Mackay**[5]
Michel Beaudouin-Lafon[6] **Robert C. Miller**[1] **Steven P. Dow**[3]

[1]MIT CSAIL
Cambridge, MA, USA
{juhokim, rcm}@mit.edu

[2]Northwestern University
Evanston, IL, USA
hq@eecs.northwestern.edu

[3]HCI Institute, CMU
Pittsburgh, PA, USA
{pandre, spdow}@cs.cmu.edu

[4]University of Washington
Seattle, WA, USA
hmslydia@cs.uw.edu

[5]INRIA
Orsay, France
mackay@lri.fr

[6]Université de Paris-Sud
Orsay, France
mbl@lri.fr

ABSTRACT
Effectively planning a large multi-track conference requires an understanding of the preferences and constraints of organizers, authors, and attendees. Traditionally, the onus of scheduling the program falls on a few dedicated organizers. Resolving conflicts becomes difficult due to the size and complexity of the schedule and the lack of insight into community members' needs and desires. *Cobi* presents an alternative approach to conference scheduling that engages the entire community in the planning process. Cobi comprises (a) communitysourcing applications that collect preferences, constraints, and affinity data from community members, and (b) a visual scheduling interface that combines communitysourced data and constraint-solving to enable organizers to make informed improvements to the schedule. This paper describes Cobi's scheduling tool and reports on a live deployment for planning CHI 2013, where organizers considered input from 645 authors and resolved 168 scheduling conflicts. Results show the value of integrating community input with an intelligent user interface to solve complex planning tasks.

Author Keywords
Cobi; conference scheduling; mixed-initiative; constraint solving; crowdsourcing; community; communitysourcing

ACM Classification Keywords
H5.2 Information Interfaces and Presentation (e.g., HCI): User Interfaces - Graphical user interfaces

INTRODUCTION

Creating a compelling schedule for a large conference is a difficult task. Hundreds of accepted submissions must be scheduled into sessions across multiple days and rooms, while accounting for the multi-faceted preferences and constraints of organizers, authors, and attendees. Organizers aim to create thematic sessions, avoid scheduling related papers or the same presenters in opposing sessions, and generally make the program interesting for attendees with different interests.

UIST'13, October 8–11, 2013, St. Andrews, United Kingdom.

ACM 978-1-4503-2268-3/13/10...$15.00.
http://dx.doi.org/10.1145/2501988.2502034

Figure 1. A small group of organizers and associate chairs create a preliminary CHI program on paper.

To better understand this challenge, we observed the schedule creation process for CHI, the largest human-computer interaction conference: CHI 2013 received over 2260 submissions and accepted more than 500 to be scheduled in 16 simultaneous sessions spanning four days. Scheduling CHI involves two stages. Once papers are accepted, a small group of associate chairs help the conference organizers to roughly create categories and suggest sessions. Over the next two days, the organizers and a few assistants build a rough preliminary schedule (see Figure 1). The process is paper-based, collaborative, and time-consuming; its output is highly dependent upon the specific knowledge of the individuals in the room. In stage two, organizers refine the rough schedule to create the final program. They attempt to resolve conflicts, handle stray papers, respond to last minute changes, and generally look for ways to improve the program. The organizers use a script to check that no presenter is scheduled to be in two places at once, but otherwise, all changes are made manually. Interviews with past organizers revealed that the process was extremely time-consuming, and that resolving conflicts was "painstaking" due to schedule complexity and the lack of feedback on whether changes resolved existing conflicts or created new ones.

Despite organizers' best intentions and efforts, previous CHI programs often contained incoherent sessions, similarly-

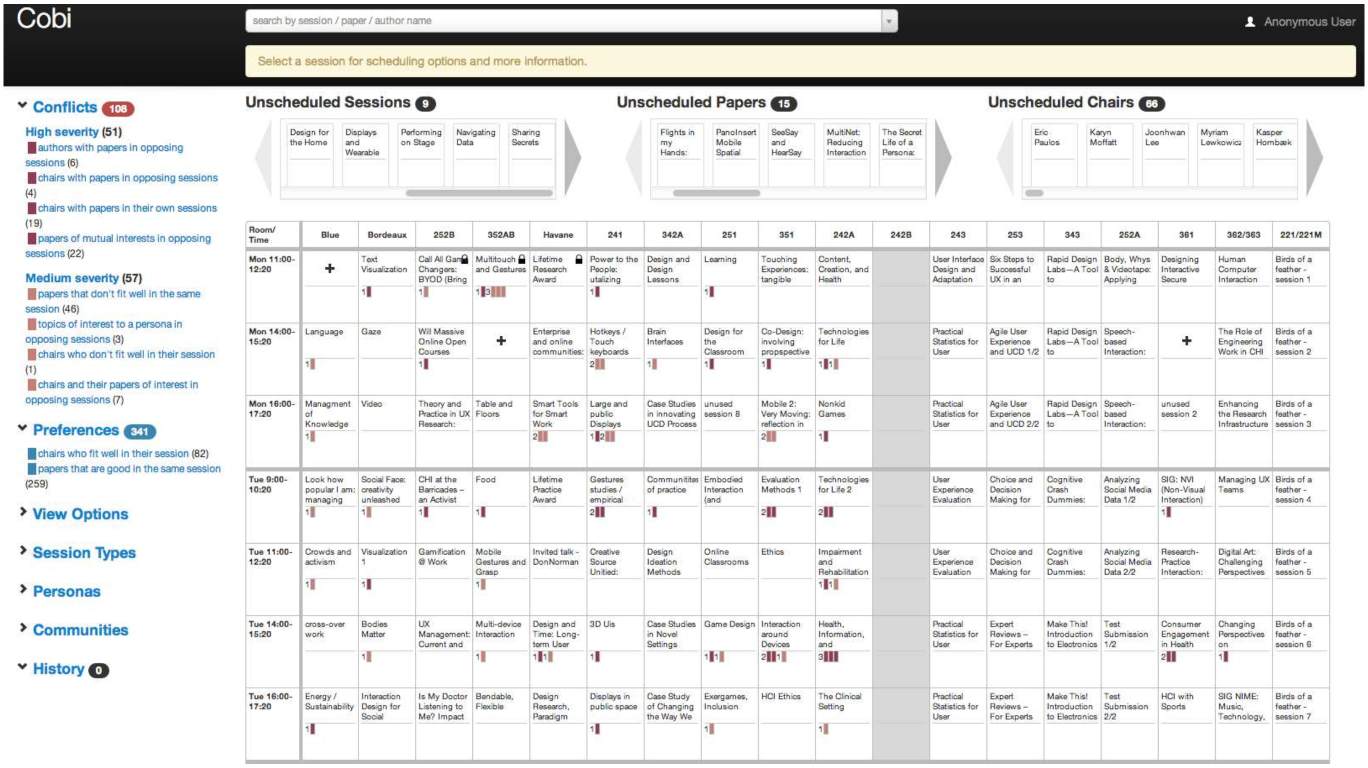


Figure 2. Cobi's scheduling tool consists of the top panel (top), the sidebar (left), and the unscheduled panel and main schedule table (right).

themed sessions that run in parallel, and author-specific conflicts. Several aspects of the process contribute to these problems. First, due to the organic nature of how organizers make connections between papers in stage one, many sessions have odd papers mixed in. Second, because the process does not capture affinities between papers in different sessions, it is difficult for organizers to make scheduling changes that lead to more cohesive sessions. Third, organizers are often unaware of the preferences of authors and attendees. This can lead to sessions of interest being scheduled at the same time. Finally, the lack of tools for managing constraints and the sheer size of the schedule make it difficult for organizers to make informed decisions when finalizing the schedule.

Cobi addresses these challenges by drawing on the people and expertise within the community, and embedding intelligence for resolving conflicts into a scheduling interface. The Cobi system consists of a collection of communitysourcing applications that elicit preferences, constraints, and affinity data from program committee members and authors, and an intelligent scheduling tool that provides organizers with helpful context and suggestions for improving the schedule. By engaging the community in the planning process, Cobi exposes the preferences and constraints of its members to the organizers and makes the planning process more transparent.

Cobi's scheduling tool (Figure 2) integrates community preferences and constraints with constraint-solving intelligence into a new kind of *community-informed mixed-initiative system*. The interface helps organizers visually spot problems and resolve them in the schedule. It highlights general, high-level conflicts such as scheduling a presenter in two opposing sessions. It also exposes more detailed, communitysourced preferences such as scheduling together papers that authors feel fit well in a session with their paper. When manipulating the schedule (e.g., assigning, moving, and swapping sessions, papers, and session chairs), the interface uses a constraint solver to help organizers make informed decisions by recommending edits that best improve the schedule and visualizing the consequences of potential edits. Organizers drive the system by applying their personal knowledge, choosing which problems to focus on, and making final decisions.

We deployed the Cobi system for planning CHI 2013. We recruited associate chairs to group sets of related papers, and authors to identify papers of interest and those that fit well in a session with their own paper. The process collected 1722 paper affinities from 64 associate chairs and 8651 preferences and constraints from 645 authors (covering 87% of accepted submissions). In addition, we asked candidate session chairs to submit their representative papers to determine their fit with sessions. The organizers used Cobi's scheduling tool to improve the preliminary schedule and assign session chairs. The tool helped the organizers resolve 168 conflicts as they created the final schedule. They found that the scheduling tool greatly simplified conflict resolution, while allowing them to combine their own knowledge with the machine intelligence and the community's input.

The paper proceeds as follows. We first discuss related work in communitysourcing and conference scheduling. We then share findings from a preliminary study and identify key design goals. We present Cobi's scheduling tool, focusing on the integration of community data, machine intelligence, and end-user interface. We report on our deployment at CHI 2013 and discuss the key lessons learned. The paper concludes with notes on future research directions.

RELATED WORK

Our work seeks to tailor tasks to the inherent incentives, interests, and expertise of diverse groups within a community. We draw from the broad literature on encouraging community contributions, both in online [10] and physical spaces [5], and on tasks ranging from collecting scientific data [4] to co-designing public transportation services [16]. In our work, community members contribute to solving a specific problem whose solution affects themselves and the community at large. With Cobi, we are exploring incentives, methods, and interfaces for collecting and incorporating multidimensional preferences and constraints from large numbers of individuals within a community into a single, cohesive outcome.

Previous research introduced mixed-initiative solutions to complex tasks such as aircraft scheduling [3] and manufacturing task scheduling [7] by modeling expert knowledge. To this line of research, Cobi contributes a community-driven approach, which raises the unique design challenges of incentivizing community members to express preferences, and mediating, encoding, visualizing, and acting on noisy and diverse community input. Interactive machine learning (IML) approaches such as CueT [1] and E-mazing [15] use a combination of human labor and machine learning. However, they take opposite approaches to the problem. IML supervises an algorithm with human input, while Cobi helps conference organizers make informed decisions with computations shown as visual feedback.

Automated scheduling is a well-studied problem in both computer science and operations research. Specific to conference scheduling, Sampson et al. [13] introduced formulations for maximizing the number of talks of interest attendees can attend. For the related problem of course scheduling, Murray et al. [11] introduced formulations for minimizing student and instructor conflicts subject to scheduling constraints. While automated scheduling is appropriate when the parameters and constraints of the optimization problem are well-specified, our interviews with past CHI organizers show that they attempt to tackle soft constraints and other tacit considerations. With Cobi's mixed-initiative, interactive optimization [6, 14], the machine plays a supporting role, providing intelligence and feedback for detecting problems and resolving conflicts. Organizers can thus better interpret and act on the community's input, which can be overwhelmingly rich, subjective, and incomplete at the same time.

Jacob et al. [8] developed a tangible interface for manipulating a conference schedule and checking constraints on a physical grid layout. Cobi extends this approach by accounting for input from the larger conference community.

There are several commercial systems for conference and course scheduling. One example is Confex's scheduling tool (`confex.com`), which detects and highlights hard conflicts in the schedule (e.g., scheduling a presenter in simultaneous sessions) but makes no suggestions about how to resolve them. Another example is UniTime (`unitime.org`), which first computes an optimal course schedule to minimize conflicts and then allows a user to make fine-grained adjustments while seeing the effect on conflicts. Our work presents an alternative approach in which the user is in control at all times while the system detects conflicts and provides suggestions for resolving them.

PRELIMINARY STUDY AND DESIGN GOALS

Our research team conducted hour-long semi-structured interviews and exchanged emails with five past and current CHI organizers, two of whom are co-authors on this paper. Discussions centered around the planning process at CHI and focused in particular on existing challenges and potential solutions. Conversations revealed three high-level goals that drove the design of the Cobi system and its scheduling tool:

Understanding paper affinities. Organizers stressed that "papers fit into sessions in complex ways" and that "getting a session together that makes sense is hard." While the in-person meeting created sessions that are mostly cohesive, organizers still needed to break open some sessions. Organizers noted that this is a "major pain point" and that it is "very hairy to break up a session" because swapping a paper with another paper requires each paper to fit well in the other's session. In order to capture paper affinities across sessions, organizers noted that you would need contributors "knowledgeable enough in the field to know that papers should or shouldn't be in the same session." Cobi draws on input from paper authors, who we hypothesize would know what other papers fit well in a session with their own.

Detecting conflicts automatically and providing feedback for resolving them. Organizers found it particularly difficult to know the consequences of moving a paper or session in the schedule, which requires reasoning about the conflicts that would be created in addition to those that would be resolved. One organizer noted that she "would (painstakingly) solve those [conflicts and] re-run [a constraint-checking script], usually showing that the problems I had solved had generated other author conflicts." In order to avoid thrashing and frustration, Cobi recommends moves and swaps that resolve the most conflicts and allows organizers to preview the effect of possible edits on conflicts.

Keeping the human in control. While an automated constraint solver can be used to resolve known conflicts, previous organizers felt that taking a purely automated approach would be impractical and would fail to capture the "many semantic constraints that are hard to express using machine understandable ways." The organizers also stressed the importance of being able to make sense of the schedule so that they can apply their knowledge and weigh the various demands of the community while scheduling.

COBI'S SCHEDULING TOOL

Once the program committee determines the accepted papers, Cobi's communitysourcing applications collect preferences, constraints, and affinity data from community members. This

Example Constraints & Preferences	*Possible Source*	*Rationale*
Papers that don't fit well together shouldn't be in the same session	Authors	Authors know what papers are related to theirs and care about which end up in a session with their own.
Papers of mutual interest shouldn't be in opposing sessions	Attendees	Knowing what attendees want to see can avoid scheduling talks of interest at the same time.
Chairs' area of research should match the topic of their session	Chairs' papers	We can collect papers from potential session chairs and check if they are related to the papers in a session.

Table 1. Examples of preferences and constraints encoded in Cobi that can be collected from community members.

input is then encoded and presented in Cobi's scheduling tool to help users (conference organizers) resolve conflicts and improve the schedule.

Encoding Preferences and Constraints

Cobi supports preferences and constraints over attributes at three entity levels: sessions, papers, and chairs. For example, a constraint may specify sessions that should not be concurrent (e.g., "sessions of interest to the ICT4D community should not oppose one another"), and a preference may state that a chair is a good fit for a session (e.g., "James Sysmaster is a good fit for the systems session"). At a high level, the goal is to create a schedule that violates few constraints and meets many preferences.

In early prototype testing, we found that most constraints and preferences of interest can be stated as conditions on a single entity or a pair of entities (e.g., "George Latewaker prefers to chair sessions in the afternoon" or "sessions on crowdsourcing and social computing should not oppose one another"). Further simplifying matters, paired-entity constraints of interest tend to describe relations over entities when they are in the same time or room, suggesting that we need only check conflicts between entities in such cases. Currently Cobi supports encoding paired entity constraints of the form "x and y should [not] be in the same session" and "w and z should [not] oppose one another," where x and y can be papers and chairs and w and z can be sessions, papers, and chairs.

Some constraints (and likewise preferences) may be *system-defined*, which refers to high-level, overarching constraints that can be stated using data from a submission management system (e.g., "a presenter should not be scheduled in opposing sessions"). Other constraints may be *community-defined*, which refers to more specific and perhaps more subjective wishes stated by community members (e.g., "Sessions A and B should be scheduled apart because Mary Liker is interested in papers in both sessions"). Table 1 provides examples of community constraints and preferences encoded in the current Cobi prototype and potential sources of community input that can be used to instantiate them.

While encoding system-defined constraints is straightforward, encoding community-provided constraints requires taking into account the potential sparsity, diversity, and subjectivity of the collected data. We may collect thousands of preferences and constraints, some of which are in direct conflict with others (e.g., an author may feel that his paper fits well in a session with another paper whose author disagrees). To account for such issues, an *input-mediation layer* aggregates responses before adding a constraint or preference in Cobi. For instance, we add a preference for two papers to be in the same session, only if the majority of people providing data about both papers agreed that they are related. We restrict two papers from being in opposing sessions only when many people express interest in seeing both papers.

In addition to managing the complexity of subjective data, the input-mediation layer can help focus the user's attention on salient constraints that matter to many community members. It can also be used to capture variance in the data and note an absence of data, so that the user can know when not to rely excessively on community input. The goal is to capture the community input at a level where the user can best act upon it: too little mediation makes it difficult to understand the community's wishes, and too much may end up hiding some of the useful information contained in the data.

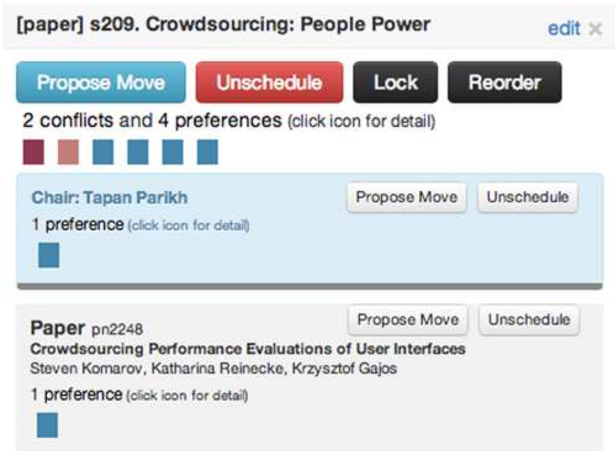
Scheduling Interface

Cobi's scheduling tool (Figure 2) enables manipulating, scanning, and reviewing the schedule with support for conflict resolution and multi-faceted views. The interface keeps the user in control and provides advice on entity moves and swaps. It consists of three components: the top panel, the sidebar, and the unscheduled panel and main schedule table.

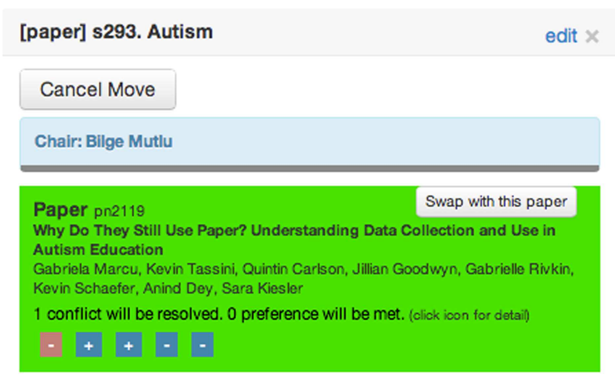
The top panel (Figure 2, top) allows the user to search for entities and displays information about the current operation. The sidebar (Figure 2, left) contains view modes and faceted browsing options to help the user analyze the current schedule. *Conflicts* and *Preferences* display conflicts and satisfied preferences in the current schedule and their counts. They are separated by type and grouped based on their severity. Counts update immediately following any change to the schedule, and provide immediate feedback on the effects of the user's actions on conflicts.

View options display different aspects of the schedule and help the user spot issues requiring attention. For example, the default *Conflict* view shows icons for conflicts that involve entities within a session. (Figure 2, right). Clicking on the *Duration* view option displays the length of each session, and allows the user to quickly identify ones with too many or too few papers. *Personas* and *Communities* allow the user to skim the schedule for various interest-based subgroups and check if the schedule is well-distributed across subgroups. The *History* option keeps track of all the scheduling operations and who performed them.

The unscheduled panel displays unscheduled sessions, papers, and session chairs. In addition to holding the entities to be scheduled, in initial testing we found that users needed



(a) View Mode



(b) Move Mode

Figure 3. Inner-session view in view mode and move mode. A recommended paper move is highlighted in green.

a scratch space to construct a session without having to worry about where it is placed in the schedule. The unscheduled panel serves as this scratch space.

The main schedule table displays the entire schedule in a time table (Figure 2, right). Each cell displays a session name and additional information based on the view option (e.g., *Conflicts*). Clicking on a session displays details of the session (Figure 3(a)), which includes conflict information and details on its papers and chair. Here the user is provided with options for scheduling entities, unscheduling entities, swapping entities, reordering papers, editing titles, and locking sessions.

When working to resolve conflicts related to an entity, the user can click *Propose Move* on a session, paper, or session chair and enter *move mode*, which displays previews of the consequences of moving to an empty slot or swapping with a candidate entity (Figure 4). Each target cell displays the net change in the number of conflicts for swapping with entities in the cell, and cells highlighted in green represent recommended moves that would lead to the largest reduction in the number of conflicts. The user can scan different options, and click on cells to examine in detail the consequences of potential moves (Figure 3(b)).

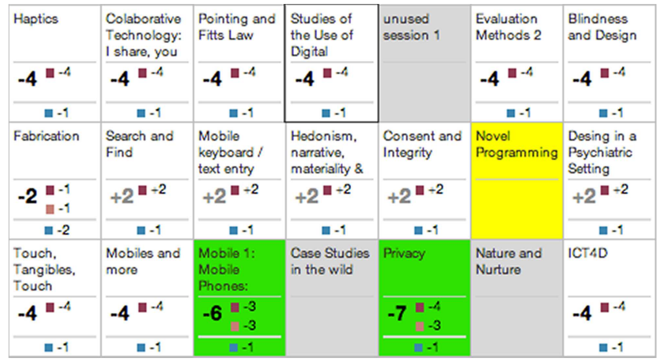


Figure 4. Move preview displays the change in the number of conflicts and preferences should the user swap the source session (shown in yellow) with each of the candidate target sessions. The system recommends sessions that minimize conflicts by highlighting them in green.

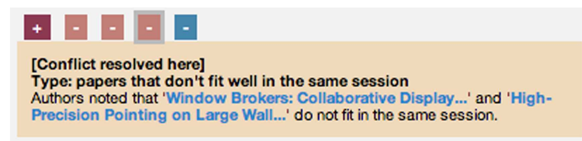


Figure 5. In move mode, conflict details preview the consequence of making a swap. By clicking on individual icons and following navigation links, the user can understand the specific conflicts and preferences that would be added or removed by making this swap.

For each empty target or target entity, Cobi displays an icon for each conflict that would be added or removed at the source and target by making the move. Clicking on the icon shows detailed information about the selected conflict or preference (Figure 5). Cobi adds links to all entities involved in the conflict or preference, so that clicking on the link highlights the selected entity. Since conflicts created or removed may also involve entities other than those being moved (e.g., a conflicting paper in an opposing session), this helps the user carefully understand which conflicts and preferences will be violated or met for which entities. Upon making a decision, the system displays the change briefly to allow the user to make sense of their decision, and returns to the view mode with updated conflict counts in the sidebar.

By using a combination of overview and detailed views, Cobi's scheduling tool aims to support quick scans as well as detailed investigations. It leaves the user in control of selecting which entities to work on and in what order, while making potential problems in the schedule evident via the conflict view and other view options. When making schedule changes, the system helps the user narrow down the set of candidates to consider, and understand visually the consequences of all possible moves on conflicts in the schedule. Since communitysourced data may be noisy or incomplete and the user may weigh various factors beyond the encoded preferences and constraints, Cobi leaves it to the user to make final scheduling decisions by applying their knowledge and making sense of recommendations from the tool.

Implementation

Cobi associates with each type of preference or constraint a lookup table containing pairs of entities that would be in conflict if particular conditions are met. Since Cobi only encodes constraints within a timeslot, conflict checking and resolution can be performed by simply taking pairs of entities in the same timeslot in the schedule and using the lookup tables to determine if they are in conflict. When computing the consequences of moves and swaps, the system uses the same lookup tables to determine which conflicts involving the source and target entities would be added or removed.

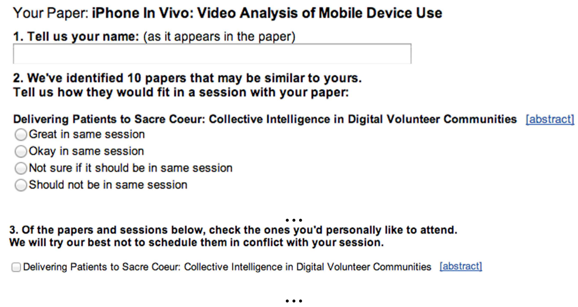
To help the user make sense of potential conflicts in the schedule, each constraint is associated with a *template message* that is instantiated with the entities in conflict when a conflict is detected. For example, a template message may state that "authors noted that x and y do not fit in the same session" and be instantiated with papers x and y that are in the same session and in the corresponding lookup table.

The scheduling tool allows multiple users to collaborate synchronously or asynchronously. The system keeps consistent transaction records on the database and pushes changes to users as they interact with the system. In cases of simultaneous, conflicting edits, the interface displays a message to the user with the failed operation, performs a local rollback, and updates with changes other users have made. Cobi also provides a polling API for other systems or interfaces to access its schedule state, which is useful when alternative visualizations or output devices are available (as in our deployment). The frontend web interface is built with HTML5, CSS3, and Javascript using the jQuery and Bootstrap toolkits.

DEPLOYMENT AND EVALUATION

We deployed the Cobi system for scheduling CHI 2013. Prior to deployment, a small group of organizers and associate chairs met in early December to produce a preliminary schedule by clustering accepted papers and making initial sessions. We deployed Cobi's communitysourcing applications between January 6 and February 12, 2013 to collect preferences, constraints, and affinity data from associate chairs, authors, and session chairs. This data was then encoded into Cobi's scheduling interface. Organizers used Cobi over a period of 42 days from February 10 to March 23, 2013. They took into account the community input, resolved session, paper, and session chair conflicts, and generally worked to improve the initial schedule.

To better understand the scheduling experience with Cobi, we collected quantitative and qualitative data from the organizers. We logged all operations the organizers executed using Cobi. A log entry includes the user responsible for the action, the action type, affected sessions or papers, and a snapshot of conflict counts as a result of the action. At the end of the deployment, we reflected on the process with each of the three CHI 2013 organizers individually (two are co-authors of this paper). Each session lasted 60-120 minutes, and was audio-recorded for later analysis. The organizers talked about high-level goals in scheduling, walked through the scheduling process, and described specific subtasks they were involved in. During the discussion they also tested the latest version of the Cobi scheduling tool that encoded the authorsourcing data, and provided feedback on the experience and usability.


Your Paper: **iPhone In Vivo: Video Analysis of Mobile Device Use**

1. Tell us your name: (as it appears in the paper)

2. We've identified 10 papers that may be similar to yours. Tell us how they would fit in a session with your paper:

Delivering Patients to Sacre Coeur: Collective Intelligence in Digital Volunteer Communities [abstract]

- Great in same session
- Okay in same session
- Not sure if it should be in same session
- Should not be in same session

...

3. Of the papers and sessions below, check the ones you'd personally like to attend. We will try our best not to schedule them in conflict with your session.

- Delivering Patients to Sacre Coeur: Collective Intelligence in Digital Volunteer Communities [abstract]

...

Figure 6. Authors were presented with a custom list of 20 papers and asked to judge which are related to their paper or of interest to them.

System-defined Preferences and Constraints

For sessions and papers, we used data from the submission management system to encode two constraints that sought to avoid scheduling paper authors in opposing sessions and sessions of interest to a persona in opposing sessions. The former constraint seeks to ensure that no presenter has to be in two places at once and that all authors can see their papers presented. The latter constraint seeks to keep sessions on a particular area of interest apart in the schedule.

For session chairs, we used data from the submission management system to encode constraints stating that chairs should not have papers in opposing sessions (for the same reason as authors), and that they should not chair sessions in which they have a paper (to avoid perceived conflict of interest).

Collecting and Encoding Community Input

We deployed three communitysourced initiatives that collected input from associate chairs, authors, and session chairs. To better understand paper affinities, we first recruited associate chairs to cluster papers in their area of expertise. The process collected 1722 paper affinities from 64 associate chairs (ACs). We then invited authors of accepted papers to identify papers that would fit well in a session with their own and that they are interested in seeing at CHI (Figure 6). To produce a small list of papers for authors to judge, we seeded suggestions based on affinities identified by ACs and by running TF-IDF (Term Frequency - Inverse Document Frequency) [9] comparisons on paper titles and abstracts. The process collected 8651 preferences and constraints from 645 authors, which covered 87% of accepted submissions. The high response rate suggests that authors were inherently interested in seeing their paper in a session with related papers and thus willing to contribute. For more information on the committee and authorsourcing stages, as well as empirical comparison of affinity creation methods, see André et al. [2].

Taking the collected authorsourcing data, Cobi's input mediation layer filtered and aggregated preferences and conflicts so that only those submitted by multiple authors were encoded. For papers, this led to encoding 923 constraints of the form "papers x and y are of mutual interests and shouldn't be scheduled in opposing sessions," 651 constraints of the form "papers x and y do not fit well in the same session," and 805 preferences of the form "papers x and y are good in the same session." For authors who also served as session chairs, we

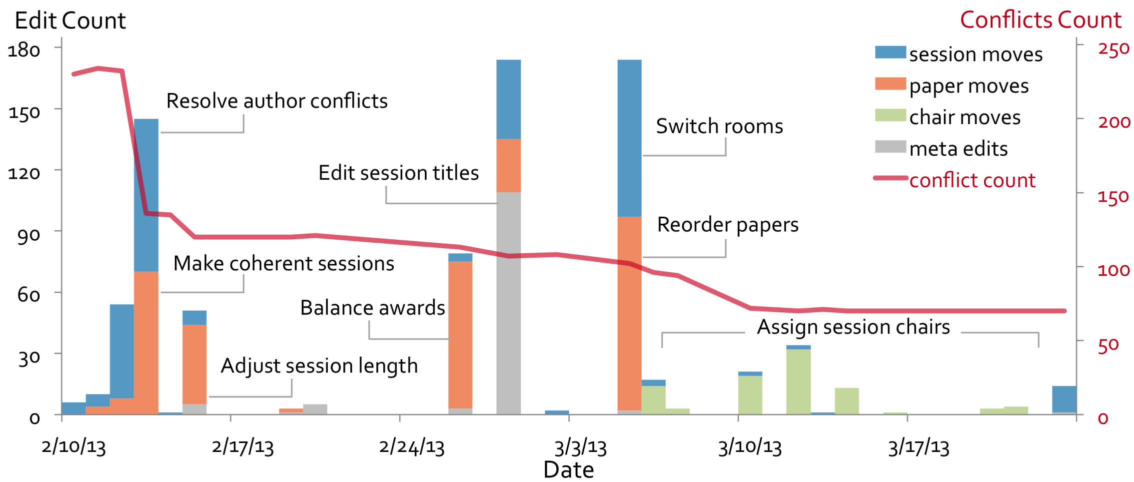


Figure 7. Representative subtasks during the CHI 2013 scheduling process are shown with associated operation types from the interface log. For example, the "Resolve author conflicts" subtask is associated with a cluster of session swap operations. Session, paper, and chair edits indicate scheduling operations involving those entities. Meta edits indicate non-scheduling operations, such as editing session titles and locking or unlocking sessions.

also added 243 constraints of the form "chair x is interested in a paper y in an opposing session." Due to time constraints in the deployment schedule, only the chair-related constraints were visible to organizers in the version of Cobi's scheduling tool they were using. For resolving other community-defined author conflicts, the organizers relied on visualizing the authorsourced data externally without the preview and recommendation support from Cobi.

In addition to the authorsourcing data, we collected representative paper samples from 165 potential session chairs, which we used to compute affinity measures on how well they may fit as chairs for a session. Using TF-IDF similarity between each session chair's papers and each session's papers, we produced affinity scores between chairs and sessions. Cobi encoded the affinity information as a preference for assigning a chair to a session when they are among the top 5 by affinity score, and a constraint when they are out of the top 25. To compute an initial assignment, we solved a linear optimization program to compute an assignment of session chairs that maximizes the sum of affinity scores.

Scheduling Process

Cobi was used in a number of scheduling meetings that involved the organizers and other collaborators. The version of Cobi that organizers used supported all session, paper, and session chair-related scheduling operations. It provided previews and recommendations for system-defined constraints, but as mentioned before, did not incorporate the authorsourcing data. For some of the meetings, Cobi was used in conjunction with a large wall display that visualized community data along with detailed session information and supported multiple users simultaneously exploring the schedule. The wall display did not include intelligence for resolving conflicts; the organizers relied on Cobi for conflict resolution and making actual changes to the schedule.

During the 42-day deployment, the three CHI 2013 organizers made 815 scheduling operations using Cobi's scheduling interface. We reconstructed the scheduling process by connecting organizers' description of the process with the interface usage log. Figure 7 shows the variety of subtasks that organizers faced during the scheduling process. By decomposing the scheduling problem into subtasks, organizers could focus on a particular aspect of the schedule at a given time.

The scheduling process proceeded in three high-level phases. In phase one (February 10 to February 17), organizers took the preliminary schedule from the technical program meeting and worked to resolve conflicts from violated system- and community-defined constraints. Organizers first moved papers so as to construct more coherent sessions based on the collected feedback from authors on which papers fit well in a session with theirs. Organizers then resolved all author conflicts by swapping sessions. While this eliminated nearly all of the existing system-defined conflicts, many sessions contained too few or too many papers. Organizers then moved papers to ensure that all sessions were at or under 80 minutes.

At the end of phase one, organizers had a mostly complete program. In phase two (February 17 to February 28), they worked to enhance themes and fine-tune the schedule to address special requirements. On the room level, they placed related sessions in the same or nearby rooms and sessions with awards in larger rooms. On the paper and session level, they distributed awards across sessions and reordered papers within a session so that they are presented in a logical progression. These subtasks generally did not involve conflict resolution, but the organizers used Cobi's conflict preview to ensure that changes would not introduce new conflicts. The organizers also made 122 session title edits to better capture and promote sessions and the papers within.

Once the organizers finalized the program, they worked in the final phase (March 1 to March 23) on assigning session chairs. Organizers first moved chairs out of sessions in which they had papers and corrected assignments where the chair was a poor fit. They then announced the initial assignments

Constraint Type	Related Entity	Data Source	Severity	Encoded	Initial	Final	Change
author with papers in opposing sessions	session (Figure 8)	system-generated	high	-	31	0	-31
topics of interest to a persona in opposing sessions	session (Figure 8)	system-generated	medium	-	6	4	-2
papers of mutual interests in opposing sessions	paper (Figure 9)	authorsourcing	high	923	40	19	-21
papers that do not fit well in the same session	paper (Figure 9)	authorsourcing	medium	651	129	42	-87
chair's paper in own session	chair (Figure 10)	system-generated	high	-	21	0	-21
chair's paper in opposing sessions	chair (Figure 10)	system-generated	high	-	6	0	-6
chair interested in opposing sessions	chair (Figure 10)	authorsourcing	medium	243	5	4	-1
chair in a session with a bad fit	chair (Figure 10)	chairs	medium	-	0	1	1
Total violated					**238**	**70**	**-168**

Preference Type		Data Source	Severity	Encoded	Initial	Final	Change
papers good in the same session	paper	authorsourcing	N/A	805	268	272	4
chair fits well in the session	chair	chairs	N/A	-	90	78	-12
Total satisfied					**358**	**350**	**-8**

Table 2. For all constraint or preference types in our deployment, the table shows the related entity, the data source for encoding, the severity level displayed in the tool, the total number of encoded items if authorsourced, the violation or satisfaction count from the preliminary schedule, the count in the finalized schedule, and the change in the count (highlighted if improved). There remains no conflicts for 3 of the 4 high severity types (highlighted). Soft constraints (medium severity) have more violations, which shows that scheduling involves multiple factors in addition to conflict resolution.

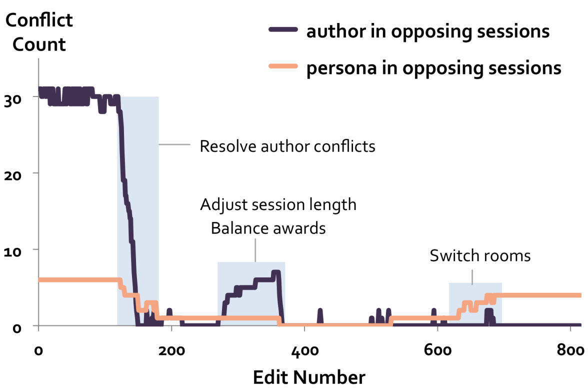


Figure 8. Change in system-defined conflicts over time. The organizers resolved 30 author conflicts during initial session making. When they were adjusting session lengths and balancing awards, the conflict count temporarily increased, but they resolved all of them shortly after. The organizers introduced three persona conflicts while they were switching rooms, but they chose not to resolve these conflicts due to other factors.

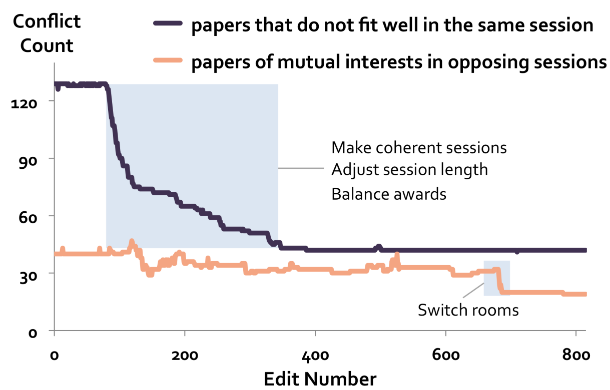


Figure 9. Change in community-defined conflicts over time. Organizers first attempted to make more coherent sessions with papers that did not fit well in the same session. After switching rooms, they resolved 11 conflicts to avoid having simultaneous sessions with similar topics.

to the session chairs, who provided additional feedback on fit and conflicts that the chairs resolved subsequently.

Conflict Resolution

The preliminary schedule from the technical program meeting included 238 conflicts. Organizers resolved 168 of them during the scheduling process. Table 2 summarizes changes in conflict counts for each constraint and preference type.

Schedule creation (phase one and two)

Organizers resolved all but four conflicts from system-defined constraints (Figure 8). Shortly after making more cohesive sessions (but in the same meeting), the organizers eliminated all 30 author conflicts in 29 minutes. This ensures that no presenter has papers in parallel sessions and that co-authors can attend all sessions that contain their papers. Persona conflicts were mostly absent because the in-person meeting to generate the initial schedule already took personas into account by scheduling sessions of interest to the same persona apart.

Organizers also resolved many of the community-defined conflicts (Figure 9). 21 of the 40 "papers of mutual interest in opposing sessions" conflicts in the initial schedule were resolved (53%), and 87 of the 129 "papers that do not fit well in the same session" conflicts were resolved (67%). Note that organizers made these changes using community-provided data and Cobi's paper-level operations, but without the previews and recommendations.

Session chair assignment (phase three)

During phase three (Figure 10), organizers resolved all 27 conflicts based on system-defined chair constraints. 6 conflicts involved chairs with papers in opposing sessions and 21 conflicts involved chairs with papers in their own session.

Reflection

Discussions with organizers revealed a number of key points on how Cobi supports the conference scheduling process and helps to resolve conflicts:

Simplifying conflict resolution with preview and feedback

The organizers commented that Cobi "trivialized conflict resolution" and was "a major stress reducer." Organizers noted

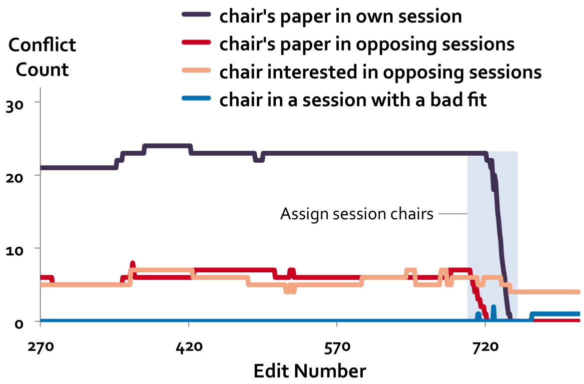


Figure 10. Change in session chair-related conflicts over time. Session chair assignment took place near the end of the scheduling process. The organizers resolved a majority of conflicts from the initial assignment by considering the individual constraints they collected from chairs.

that Cobi's visual previews aided in understanding the consequence of making a move, and immediate feedback accelerated conflict resolution. One organizer described the experience of resolving system-defined conflicts as follows: "It went really really fast, because we see a session that has conflicts, and the suggestions Cobi was giving were really good, and then we swapped things. It was almost a no brainer."

Enabling mixed-initiative problem solving

Cobi allowed organizers to use constraint-solving intelligence alongside their own knowledge of the schedule and of tacit constraints to improve the program. One organizer commented that "I was by and large driven by what Cobi was suggesting. As you make progress you can then progressively integrate other criteria that are not explicit in the system." In swapping sessions to resolve author conflicts, another organizer noted that he used Cobi's conflict previews to find good sessions to swap with that were in the same room, so as to resolve conflicts while maintaining the themes that were loosely assigned to rooms during the in-person meeting. Cobi also allowed organizers to be aware of causing potential conflicts even when they weren't working to resolve them. "We had Cobi up all the time to make sure that when we had a solution that we thought worked from the affinity point of view, that it didn't introduce new conflicts."

An important aspect of any mixed-initiative system is the user's trust in system recommendations. Organizers noted that their trust in Cobi grew over time. "For those of us who returned to the problem on multiple occasions, with a diverse set of short-term colleagues with varying expertise who came in to help, I think our appreciation of Cobi actually grew. We were looking at C&B [contribution & benefit] statements, as well as abstracts, and double-checking with our visitors, and Cobi kept coming up with great suggestions. So it stopped being based on following Cobi because it dealt with the areas that we knew, and became more fundamental, because it held up under scrutiny from a variety of visitors and when we delved into the details of various papers."

Intelligence powered by community input

Organizers commented on the value of having authorsourcing data during the scheduling process. In previous years, "authors saw their paper move from a slot they liked to a time they didn't and talked to the TP chairs." But this year, "we had virtually none of this. Authors were asked for input, most gave it, we tried hard to accommodate them, and almost nobody complained."

Organizers noted that authorsourcing data was particularly useful for understanding the affinity among papers beyond the initial sessions created at the TP meeting. "It helped with the more subtle issue of what happens when a paper moves out of a 'happy session'. Pairs of papers with a strong affinity are not in conflict if in the same session, and become conflicted if they move out of that session, but stay in the same time slot. This got lost over and over again in previous years and is the big win this year."

The organizers used a wall display to visualize authorsourcing data. Since the version of Cobi that organizers used at the time did not incorporate this data, organizers had to manually identify paper swaps that lead to more cohesive sessions. Later, when asked to test a version of Cobi's scheduling tool that encoded authorsourced preferences and constraints, organizers noted that having Cobi able to propose and resolve community-defined conflicts would have been valuable: "That is precisely what I was doing by hand with the authorsourcing data. And [the blue icon showing community preference] is absolutely useful." "I think if we had this earlier when we were doing the first round of improving the sessions, we would certainly have used it."

Weighing priorities and deliberately leaving conflicts

If Cobi makes it easy to resolve conflicts, why does the final schedule still include 70 conflicts? When asked this question, an organizer replied that for many of the remaining conflicts "we felt that they were minor or we decided that they weren't really conflicts." Since there are multiple factors that affect scheduling decisions, satisfying all constraints is sometimes not possible. An organizer commented that "it's also a question of opinion in some cases." In these cases the organizers made the final decision, which corresponds to Cobi's design goal to support decision-making while keeping the user in control at all times.

Mediating and visualizing community input

While the authorsourcing data provided a rich perspective on the community's preferences and constraints, the data was also sparse and noisy. In resolving authorsourced conflicts, organizers strived to understand the complexity of community input. Organizers used their wall display to visualize the raw community data, and attempted to account for the variance, quality, and weight of the data when making decisions. While organizers appreciated Cobi's aggregating authorsourcing data to remove noise and highlight salient conflicts, they also noted that visualizing the raw data was helpful and gave them more flexibility in using community input. Based on this feedback, we plan to develop other methods for mediating and visualizing community input, that can reflect its variance and quality while also maintaining simplicity.

Visualization improvements

Organizers noted a few areas for potential improvement in Cobi's visualization. A major issue raised was that Cobi can only display the details of one session at a time. While scheduling, organizers found that they often wanted to compare multiple entities. One organizer noted that "having to sequentially navigate multiple items worsens the experience a bit." To address these comments, we are currently exploring alternative visualizations for displaying detailed information for multiple sessions and are considering applying focus plus context techniques such as those found in TableLens [12]. The challenge is in providing additional context without distorting the schedule table in ways that hinder sensemaking.

Another solution to the visualization challenge is to leverage more space when available. For example, the wall display the organizers used for planning CHI 2013 was large enough to visualize detailed submission information in the global view. It allowed a group of people to collaborate on scheduling subtasks, although it lacked conflict resolution capability. A possible extension for Cobi is to directly connect to a large display so as to facilitate collaboration and enable the micro-outsourcing of scheduling tasks.

CONCLUSION AND FUTURE WORK

The Cobi system integrates community process, constraint-solving intelligence, and end-user interface to help organizers plan large conference schedules. Cobi's scheduling tool encodes community input as preferences and constraints, and helps organizers resolve conflicts by providing previews and recommendations when editing the schedule. A live deployment of Cobi for planning CHI 2013 demonstrated the effectiveness of collecting preferences and constraints from community members, and of the scheduling tool for simplifying conflict resolution and supporting informed decision-making.

The challenges of conference scheduling—understanding paper affinities, knowing what people want, and managing the solution complexity—are shared by conferences beyond CHI. We believe the approach presented in this paper generalizes to scheduling other academic conferences, and also to other events such as trade shows, film festivals, or university courses. More generally, the success of Cobi's deployment posits community-informed, mixed-initiative interaction as a novel approach for solving optimization problems, for which the goal is to collect important data from the community, use computation to guide the solution, and allow users to apply their tacit knowledge.

In future work, we plan to explore ways to further engage the community in the scheduling process. We wish to provide a generalized method for community members to express arbitrary constraints and preferences (e.g., travel plans and special considerations). We are currently working on an interactive interface that allows community members to specify a broader range of preferences and constraints. The collected input can then be encoded like other community-defined constraints, so that conflicts can be easily resolved using Cobi.

Another direction for future work is to extend the role of the community beyond providing data. Can hundreds of people collaboratively make sessions and resolve conflicts? We imagine that tools can support new ways of communicating, collaborating, and incorporating different opinions. Pursuing this research direction can lead to a new family of community-supported mixed-initiative systems that better mediates and visualizes diverse input from the community.

ACKNOWLEDGMENTS

We thank Patrick Baudisch, Kia Höök, Ed Chi, Pedro Lopes, Nirmal Patel, and members of the CHI community. This work is supported in part by Quanta Computer as part of the Qmulus project, by the Ford-MIT Alliance, and by NSF under awards SOCS-1111124 and SOCS-1208382. Juho Kim is partly supported by the Samsung Fellowship.

REFERENCES

1. Amershi, S., Lee, B., Kapoor, A., Mahajan, R., and Christian, B. Cuet: human-guided fast and accurate network alarm triage. In *Proc. CHI'11* (2011), 157–166.
2. André, P., Zhang, H., Kim, J., Chilton, L. B., Dow, S. P., and Miller, R. C. Community clustering: Leveraging an academic crowd to form coherent conference sessions. In *Proc. HCOMP 2013, in press* (2013).
3. Butler, K. A., Zhang, J., Esposito, C., Bahrami, A., Hebron, R., and Kieras, D. Work-centered design: a case study of a mixed-initiative scheduler. In *Proc. CHI'07* (2007), 747–756.
4. Evans, C., Abrams, E., Reitsma, R., Roux, K., Salmonsen, L., and Marra, P. P. The neighborhood nestwatch program: Participant outcomes of a citizen-science ecological research project. *Conservation Biology 19*, 3 (2005), 589–594.
5. Heimerl, K., Gawalt, B., Chen, K., Parikh, T., and Hartmann, B. Communitysourcing: engaging local crowds to perform expert work via physical kiosks. In *Proc. CHI'12* (2012), 1539–1548.
6. Horvitz, E. Principles of mixed-initiative user interfaces. In *Proc. CHI'99* (1999), 159–166.
7. Hsu, W.-L., Prietula, M. J., Thompson, G. L., and Ow, P. S. A mixed-initiative scheduling workbench: integrating ai, or and hci. *Decis. Support Syst. 9*, 3 (Apr. 1993), 245–257.
8. Jacob, R. J. K., Ishii, H., Pangaro, G., and Patten, J. A tangible interface for organizing information using a grid. In *Proc. CHI'02* (2002), 339–346.
9. Jones, K. S. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation 28*, 1 (1972), 11–21.
10. Kraut, R., and Resnick, P. *Building Successful Online Communities: Evidence-Based Social Design*. Mit Press, 2011.
11. Murray, K., Müller, T., and Rudová, H. Modeling and solution of a complex university course timetabling problem. In *Proceedings of the 6th international conference on Practice and theory of automated timetabling VI* (2007), 189–210.
12. Rao, R., and Card, S. K. The table lens: merging graphical and symbolic representations in an interactive focus + context visualization for tabular information. In *Proc. CHI'94* (1994), 318–322.
13. Sampson, S. E. Practical implications of preference-based conference scheduling. *Production and Operations Management 13*, 3 (2004), 205–215.
14. Scott, S. D., Lesh, N., and Klau, G. W. Investigating human-computer optimization. In *Proc. CHI'02* (2002), 155–162.
15. Stumpf, S., Sullivan, E., Fitzhenry, E., Oberst, I., Wong, W.-K., and Burnett, M. Integrating rich user feedback into intelligent user interfaces. In *Proc. IUI'08* (2008), 50–59.
16. Zimmerman, J., Tomasic, A., Garrod, C., Yoo, D., Hiruncharoenvate, C., Aziz, R., Thiruvengadam, N. R., Huang, Y., and Steinfeld, A. Field trial of tiramisu: crowd-sourcing bus arrival times to spur co-design. In *Proc. CHI'11* (2011), 1677–1686.