Social Event Organization: An Example

You want to organize the annual corporate day, and plan activities/events that the employees will enjoy with one another.
Common characteristics of Social Event Organization

Participants may have preference for events

Participants may want to be with friends

Events have constraints on number of participants (min. and max.)
Social Event Organization: Problem

Given a set of users, and a set of events,
  <user, event> innate affinity,
  <user, user> social affinity,
  cardinality constraints on events (*lower and upper*)

Find an assignment of users to events that maximizes the total utility (say, a linear function of the two affinities)
Example

<table>
<thead>
<tr>
<th>User</th>
<th>Event</th>
<th>Affinity</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Chess</td>
<td>0.8</td>
</tr>
<tr>
<td>B</td>
<td>Poker Games</td>
<td>0.7</td>
</tr>
<tr>
<td>C</td>
<td>Poker Games</td>
<td>0.4</td>
</tr>
</tbody>
</table>

Chess: min 2 max 2
Poker Games: min 2 max 9

Assignment 1
Chess: A
Poker Games: B, C
Chess will not materialize

Assignment 2
Poker Games: A, B, C

How does this dynamics change when social affinity is added?
Key Related Work

1. **Generalized Assignment Problem (GAP)**
   - **Problem**: assign items to bins respecting capacity to maximize profit
   - **Difference**: social affinity not considered

2. **National Resident Matching Program (NRMP)**
   - **Problem**: two-sided matching of graduates to residency programs
   - **Difference**: social affinity not considered, stability is important

3. **Team formation**
   - **Problem**: assign users to teams to fulfill skill requirement, with minimum communication overhead
   - **Difference**: no cardinality constraints

4. **Group recommendations**
   - **Problem**: recommend items to a group of users that maximize utility
   - **Difference**: groups known in advance
Problem Definition - Social Event Organization (SEO)

users $U$ and events $A$

innate affinity $\sigma : U \times A \rightarrow \mathbb{R}_+$

social affinity $w : U \times U \rightarrow \mathbb{R}_+$

lower and upper cardinality $\gamma_a, \delta_a, \forall a \in A$

utility of a set of users $S \subset U$ assigned to event $a \in A$

$$\text{utility}(S, a) = (1 - \alpha) \sum_{u \in S} \sigma_{u,a} + \alpha \sum_{u,v \in S, u \neq v} w(u, v)$$

an assignment $M : U \rightarrow A$

set of users assigned to the event $M^{-1}(a) = \{u \in U | M(u) = a\}$

The goal is to find the assignment that maximizes the overall utility

$$M^* = \text{arg max}\{\sum_{a \in A} \text{utility}(M^{-1}(a), a) \mid M \text{ is feasible}\}$$
Hardness Results

Theorem 1: The SEO Problem is NP-complete to approximate within any constant factor.

Theorem 2 (Special Case: Innate Utility only): It is NP-hard to approximate the innate-only restricted version of SEO within a factor of \((1-1/n+\varepsilon)\) in polynomial time, where \(n\) is number of users.

Theorem 3 (Special Case: Social Utility only): It is NP-hard to approximate the social-only restricted version of SEO within any constant factor in polynomial time.
Solution Terminology

During the assignment process, let $S_a$ be the set of users (tentatively) assigned to event $a$, then

- event $a$ is Phantom if $|S_a| < \gamma_a$
- event $a$ is Real if $|S_a| \geq \gamma_a$
- event $a$ is Open if $|S_a| < \delta_a$ else Closed
- user $u$ is Available if not assigned to a Real event

Recall: A solution is *feasible* if all cardinality constraints of all non-empty events are satisfied
Solution Template

1. Compute L, a priority queue of potential <user, event> assignments with scores

2. Make one pass over L to execute assignments based on status of users (availability) and events (real/phantom, open/closed).
   - When event a becomes real, invalidate any past and future assignments of users in $S_a$

3. Perform post-processing for phantom event assignments

Several orderings possible

Ordering may be updated at each assignment

Each assignment impacts future utility
Greedy Baselines

Static Greedy
Sort L based on pairwise gain

\[ g((u,v), a) = (1 - \alpha)(\sigma_{u,a} + \sigma_{v,a}) + 2\alpha w(u, v) \]

Dynamic Greedy
Update L at each assignment based on current assignments

\[ g(u, a \mid S_a) = (1 - \alpha)\sigma_{u,a} + \alpha \sum_{v \in S_a} w(u, v) \]

Can do better by accounting for potential gain from future assignments
Greedy Solution with Look Ahead Estimation

**Phantom Aware:** Limits the creation of new phantom events and encourages filling current spots by maintaining a *deficit* variable.

**Community Aware:** For any assignment \((u,a)\), estimate the gain in social utility achieved by remaining assignments to that event.

Together called the Phantom and Community Aware Dynamic Greedy (PCADG)
Evaluation on real dataset

Datasets:
- Meetup: NYC, SFO
- Plancast: VAN, CHI
- Sigcomm2009

SFO: 6438 users and 59 events
NYC: 10328 users and 127 events
VAN: 2338 users and 339 events
CHI: 2327 users and 360 events
Sigcomm2009: 76 users and 11 events
Evaluation w.r.t. Total Utility

**PCADG**: Phantom and Community Aware Dynamic Greedy
**PADG**: Phantom Aware Dynamic Greedy
**DG**: Dynamic Greedy
**SG**: static greedy

2x-4x improvement when social network is considered
Evaluation w.r.t. Regret Ratio

Regret Ratio: Coarse upper bound w.r.t. max achievable utility

\[ \rho(u) = 1 - \frac{(1 - \alpha)\sigma_{u,M(u)} + \alpha \sum_{v \in S_{M(u)}} w(u, v)}{\max_{a \in A}((1 - \alpha)\sigma_{u,a} + \alpha \sum_{v \in B_u} w(u, v))} \]

where, \( B_u \in N_u \) is the top \( K_{u,a} = \min\{|N_u|, \delta_a - 1\} \) friends of \( u \).

Note it’s a measure on user-level.
Evaluation w.r.t. Regret Ratio

PCADG still consistently outperforms all baselines
Closing thoughts

- Defined the problem of social event organization
- Incorporating social affinity yields significant gains in utility
- The coupling effect across users makes dynamics of social systems very complex.

- What if social and innate affinities are interdependent?
- Do other utility functions yield exact or approximable solutions?
The end. Thanks.
Evaluation on synthetic dataset

Effects of parameter choices on the performance of PADG and PCADG

Running time (in seconds) of PADG and PCADG
Varying contributions of social and innate affinities

Dominance of PCADG is still preserved