Online Policies for Efficient Volunteer Crowdsourcing

How can we use nudging mechanisms to engage volunteers efficiently while avoiding excessive notifications?

Motivated by a collaboration with FOOD RESCUE US (FRUS)

The Simple Solution To Ending Local Hunger

Our Contributions:

- Introduce the online volunteer notification problem
- Develop online policies with constant factor guarantees
- Provide hardness results
- Test policies on datasets from FRUS

Authors: Vahideh Manshadi and Scott Rodilitz (Yale School of Management)

Paper: https://arxiv.org/abs/2002.08474

Online Volunteer Notification Problem

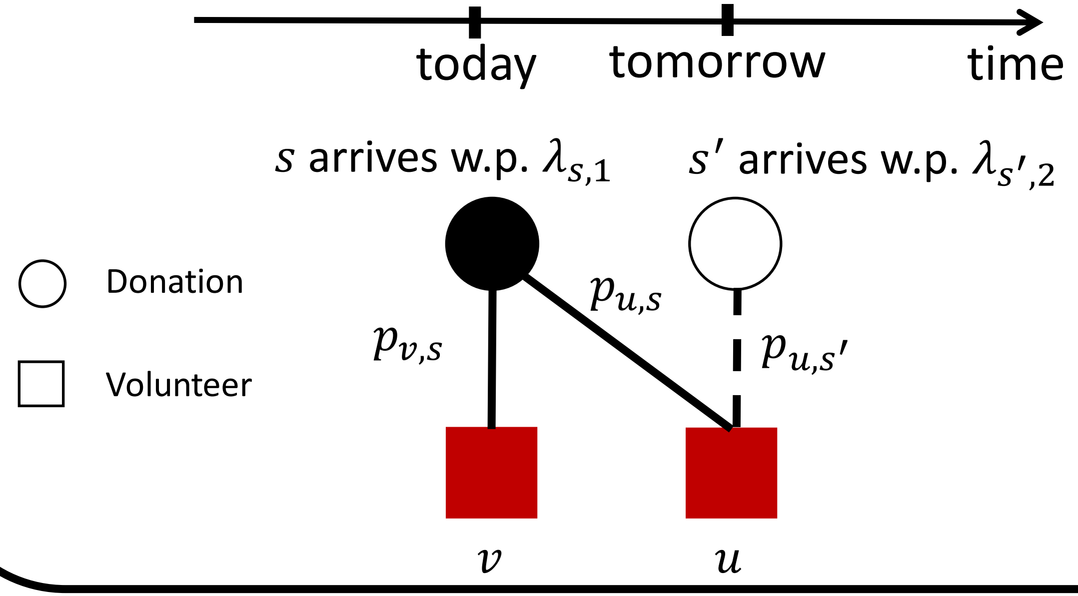
- V: set of volunteers and S: set of task types
- **Task arrival:** At time t task s becomes available w. p. $\boldsymbol{\lambda_{s,t}}$ $(\sum_{s\in S}\lambda_{s,t} \leq 1)$
- **Volunteer state (active/inactive):** Initially active.
- **Match prob.:** If active & notified about s, volunteer v responds w. p. $\boldsymbol{p_{v,s}}$
  - If an active volunteer is notified, she becomes inactive for τ periods
- **Inter-activity time distribution:** $g(\tau)$

Challenging Objective: maximize expected # of completed tasks over T periods (submodular in notified subset)

Goal: design online notification policies that perform well compared to a "clairvoyant benchmark"

Example:

We can notify each volunteer every two days ➡ $g(2) = 1$



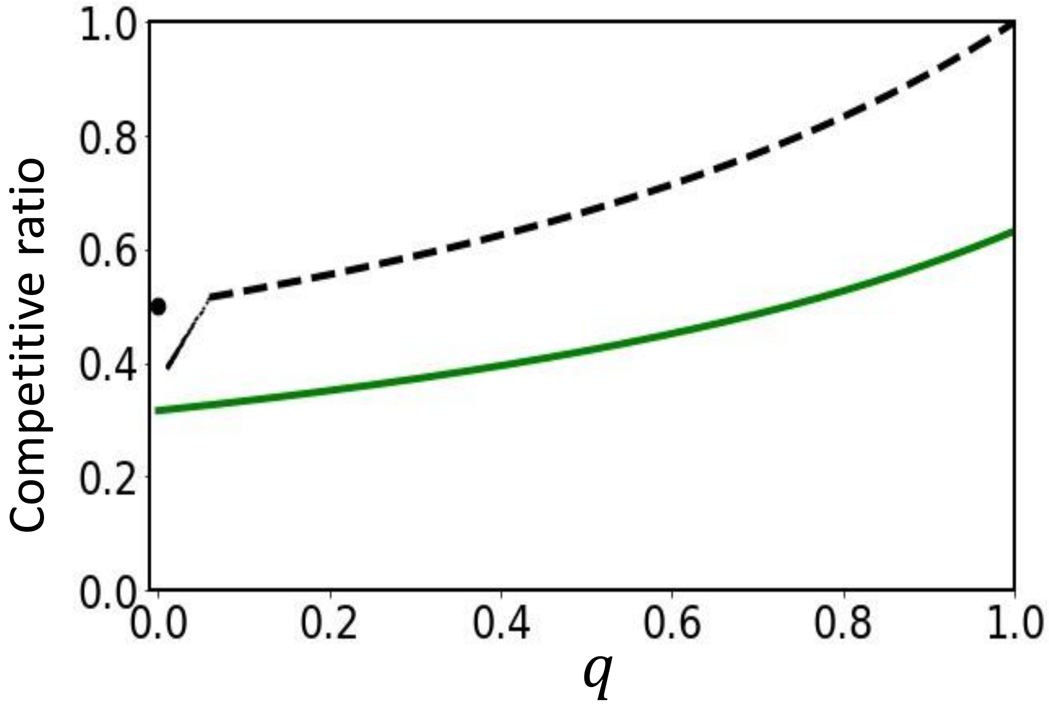
Summary of Theoretical Results

- Parameterized based on minimum discrete hazard rate of inter-activity time distribution:

$$q = \min_{\tau} \frac{g(\tau)}{1 - G(\tau - 1)}$$

Theorem [Lower Bound]: There exists a non-adaptive randomized online policy that achieves at least $(1 - {}^{1}/_{e})\ \frac{1}{2-q}$ of our benchmark.

Theorem [Upper Bound]: If $q = 1/n$ for some integer n, no online policy can achieve better than $min\left\{\frac{1}{2-q}, 1 + q + \frac{q(1-q)(1-e^{-1})}{(1+q)\log(1-q)}\right\}$ of our benchmark.



Sparse Notification Policy (SNP)

Key Idea: Sparsify an ex-ante solution $\boldsymbol{x}^*$ by solving a sequence of "low-dimensional" DPs

Offline Phase: Artificially rank volunteers. Starting with $v = 1$ and $t = T$,

$$y_{v,s,t} = \begin{cases} x^*_{v,s,t} & \text{Reward of notifying } v \text{ about } s \text{ at } t \geq \text{Reward of not notifying } v \text{ at } t \\ 0 & \end{cases}$$

Reward of notifying v about s at t:

$$p_{v,s} \prod_{1 \le u \le v-1} \left(1 - p_{u,s}\, y_{u,s,t}\right) + \sum_{t+1 \le \tau \le T} g(\tau - t)\, \boldsymbol{J_{v,\tau}}$$

Reward of not notifying v at t: $J_{v,t+1}$

$y_{u,s,t}$: Solution of higher ranked DP's

$J_{v,\tau}$: Expected future number of rescues completed by v (if active at τ)

$$J_{v,t} = \sum_{s \in S} y_{v,s,t} \text{ (Reward of notifying } v \text{ about } s \text{ at } t\text{)}$$

$$+ (1 - y_{v,s,t}) \text{ (Reward of not notifying } v \text{ at } t\text{)}$$

Online Phase: If task s arrives at time t, notify volunteer v with prob. $y_{v,d,t}$

SNP vs. FRUS Current Practice