Mobile Crowdsourcing: Incentives, Trust, and Privacy

IEEE ICC 2016 Tutorial

23rd May 2016

Salil Kanhere, The University of New South Wales, Australia

http://www.cse.unsw.edu.au/~salilk

Tony T. Luo, Institute for Infocomm Research, A*STAR, Singapore

http://www1.i2r.a-star.edu.sg/~luot/

Download link for the latest slides:

http://www1.i2r.a-star.edu.sg/~luot/download/ICC16_T4.pdf

Outline

- Introduction (30')
- Incentives (60')
  - Fundamentals of mechanism design
    - Bayesian mechanism design
  - Crowdsourcing and All-pay auctions
  - Tullock contests

--- Break ---

- Trust (40')
  - Motivating experiments
  - Reputation framework
  - A social-network perspective

- Privacy (40')
  - Collaborative path hiding
  - AnonySense: Anonymous Tasking and Reporting
  - Private Data Vaults: Access Control
  - IncogniSense: Balancing Privacy and Trust

- Summary and Conclusions (10')

INTRODUCTION

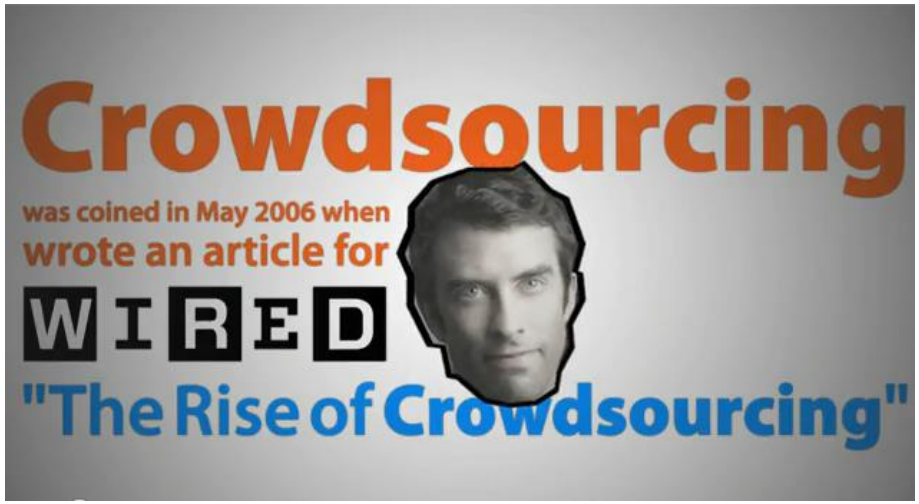
Crowdsourcing

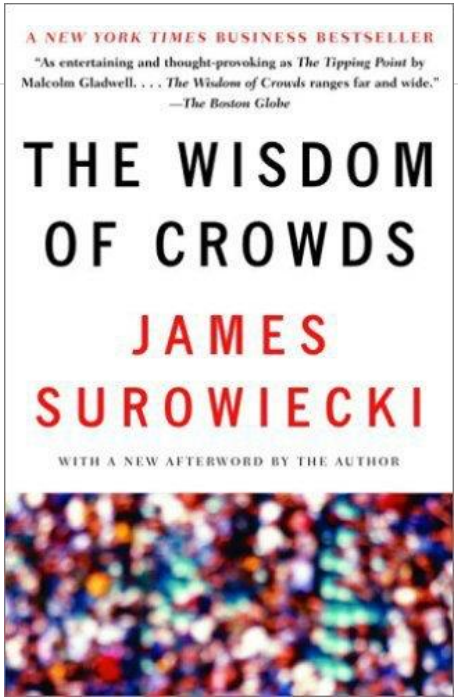
crowdsource | ˈkraʊdsɔːs |
verb *[with obj.]*

obtain (information or input into a particular task or project) by enlisting the services of a number of people, either paid or unpaid, typically via the Internet: *she crowdsourced advice on album art and even posted an early version of the song so fans could vote for their favorite chorus* | (as noun **crowdsourcing**) *: the paper seems more comfortable than many of its rivals wading into the world of crowdsourcing and citizen journalism.*

ORIGIN

early 21st cent.: from CROWD + SOURCE, after OUTSOURCE.





1714

A Journey Through Time

Measurement of longitude was a problem during transoceanic voyages

British government established the Longitude Act which offered substantial monetary reward for solutions:

- £10,000 for a method that could determine longitude within 1 degree (110km at equator)
- £15,000 for a method that could determine longitude within 40 minutes
- £20,000 for a method that could determine longitude within 30 minutes

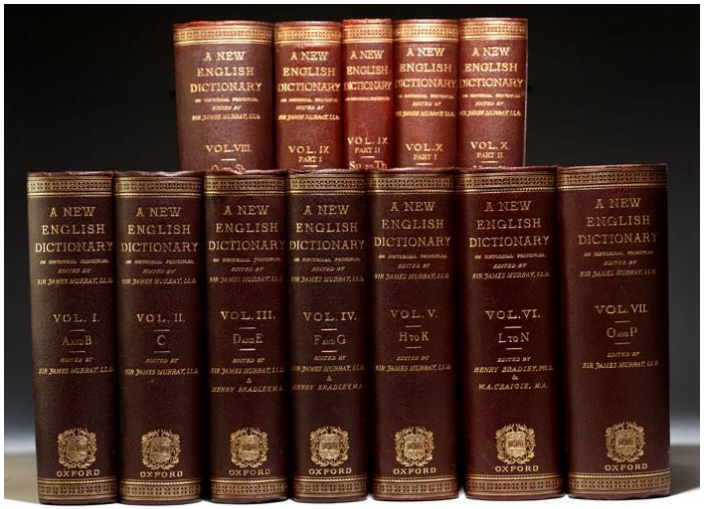
1884

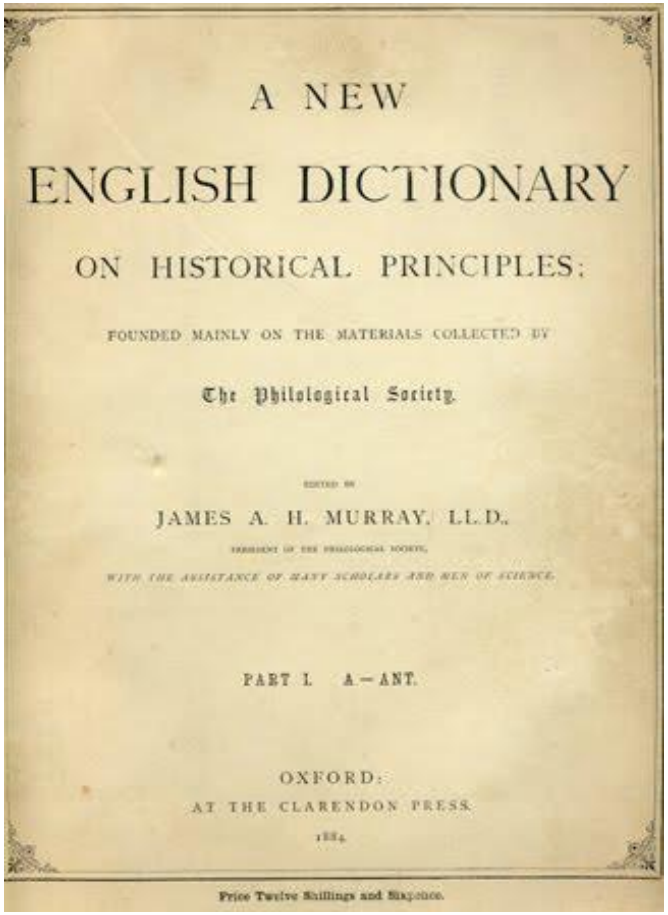
A Journey Through Time

Publication of the Oxford English Dictionary

800 volunteers read old manuscripts and catalogued words to create the first fascicle

Took 70 years to complete




A NEW
ENGLISH DICTIONARY
ON HISTORICAL PRINCIPLES:
FOUNDED MAINLY ON THE MATERIALS COLLECTED BY
The Philological Society.
EDITED BY
JAMES A. H. MURRAY, LL.D.,
PRESIDENT OF THE PHILOLOGICAL SOCIETY,
WITH THE ASSISTANCE OF MANY SCHOLARS AND MEN OF SCIENCE.
PART I. A—ANT.
OXFORD:
AT THE CLARENDON PRESS.
1884.
Price Twelve Shillings and Sixpence.

1970

A Journey Through Time

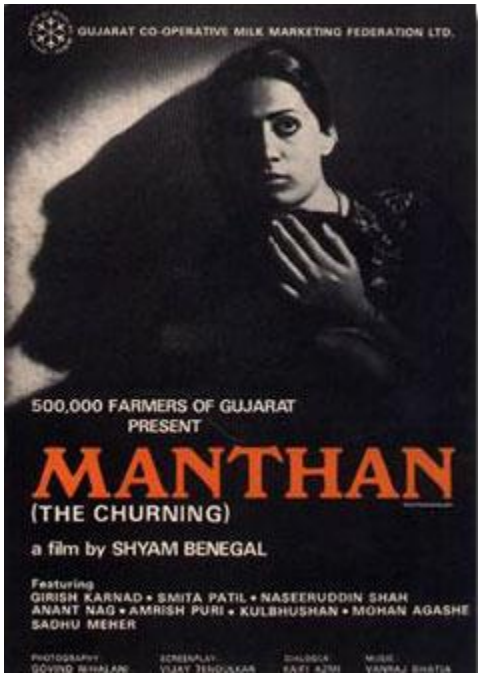
“This was Paris in 1970” amateur photo contest

Over 14,000 photographers contributed 70,000 B&W prints and 30,000 colour slides of Paris to document the architectural changes in the city

1976

A Journey Through Time

A Hindi film made by Shymal Benegal, based on the story of the pioneering milk cooperative movement of Vergese Kurien



First film in the world to be crowd-funded. Over 500,000 famers of the Gujrat Co-operative Milk Federation contributed Rs.2 each

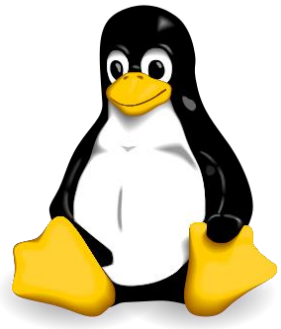
1980-90's

A Journey Through Time

1983: Richard Stallman published the GNU Manifesto and launched GNU Project to write an open-source OS

1989: First version of GNU GPL

1991: Linus Torvalds released the Linux kernel



1998: “Open source” label created shortly after the release of the Netscape source code

2000's

A Journey Through Time

2000: iStockPhoto, online free stock imagery website where the public can contribute photos and receive commission

2001: Wikipedia

2005: Amazon Mechanical Turk

2008: StackOverflow

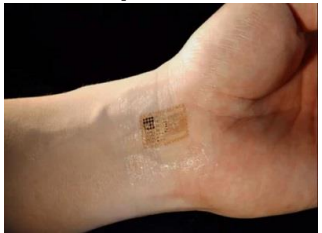
2009: TaskRabbit

Crowdsourcing goes Mobile

Smartphones combine sensors, computation and communication

- accelerometer
- gyroscope
- magnetometer
- front and rear cameras
- NFC
- barometer
- speaker
- microphone
- proximity
- light sensor
- Bluetooth
- GPS
- WiFi + cellular
- humidity
- temperature

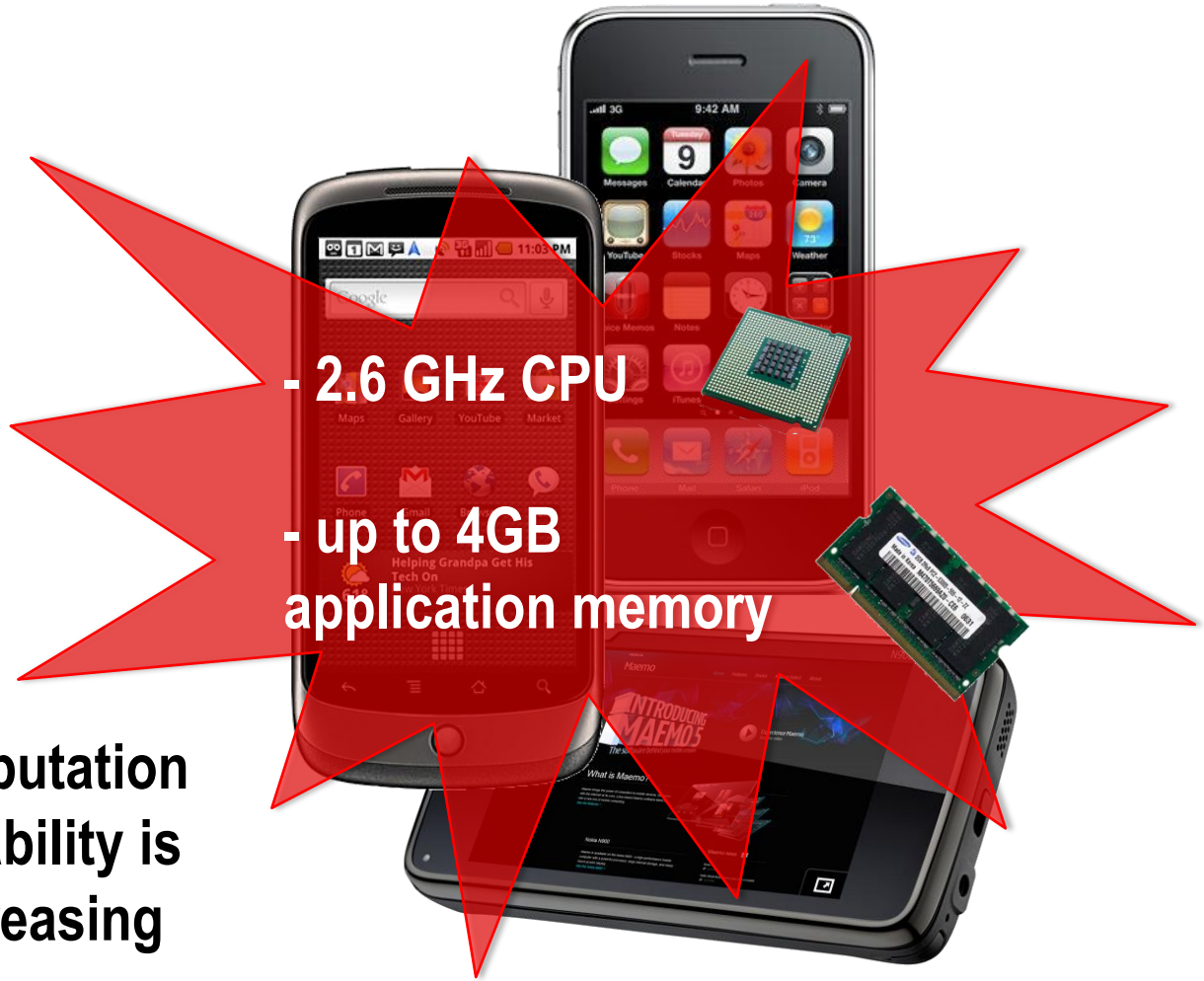
Plethora of external sensor can speak wirelessly with smartphones



Density

Programmability

Hardware



computation capability is increasing

Application Distribution

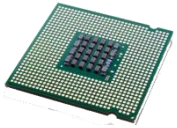
deploy apps onto millions of phones at the blink of an eye

Cloud Infrastructure

- sensing
- run machine learning algorithms locally (feature extraction + inference)

programmability

sensing



App Store

ovi NOKIA

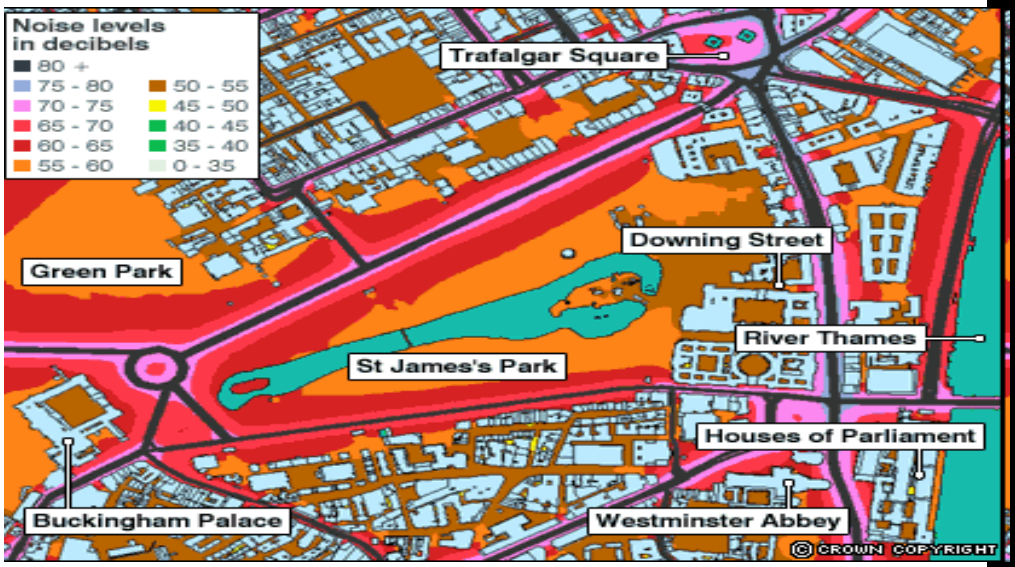
cloud infrastructure


Societal scale sensing
a global mobile sensor network

Application Domains

Mobile Crowdsourcing: Environment

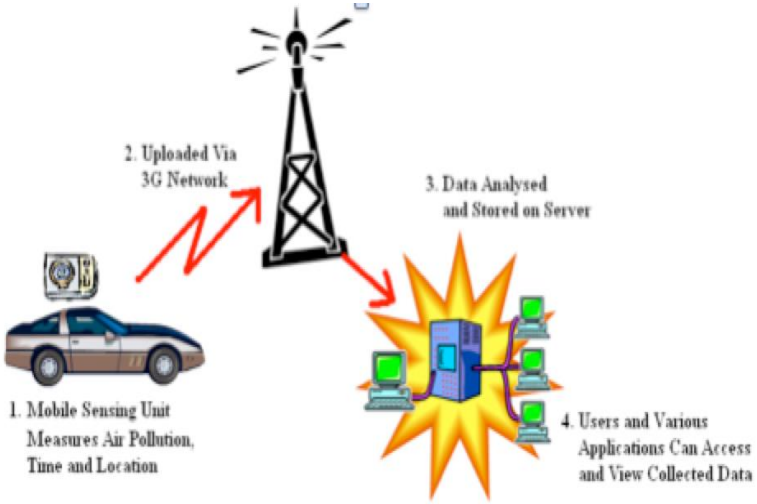
EarPhone: Noise Pollution Mapping

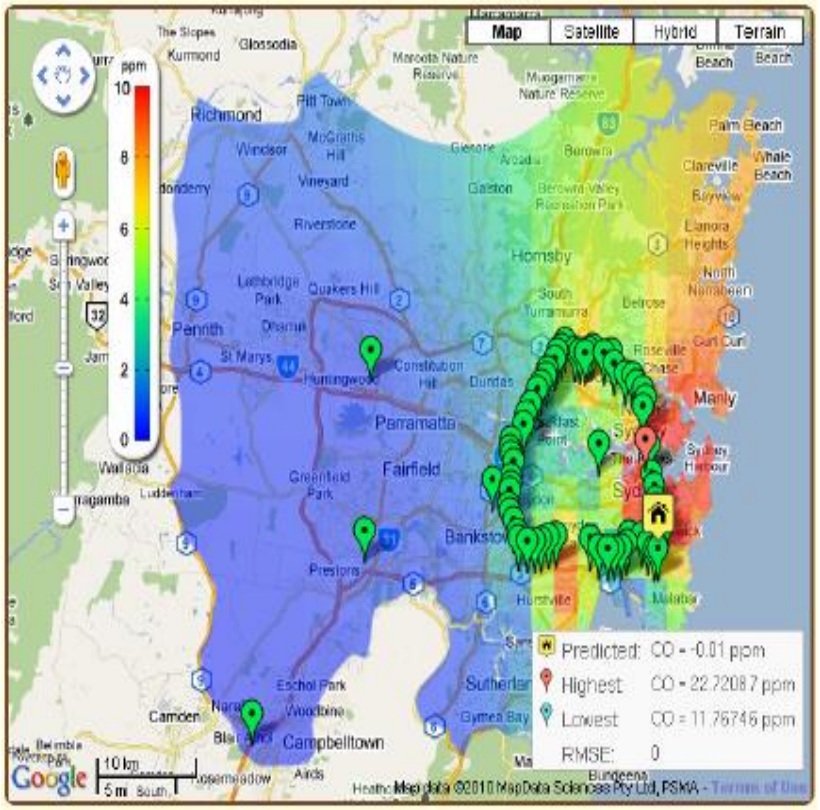


R. Rana. C. T. Chou, S. S. Kanhere, W. Hu and N. Bulusu, “Earphone: An End-to-End Urban Participatory Urban Noise Mapping System”, in Proceedings of ACM IPSN, Stockholm, Sweden, April 2010.

Mobile Crowdsourcing: Environment

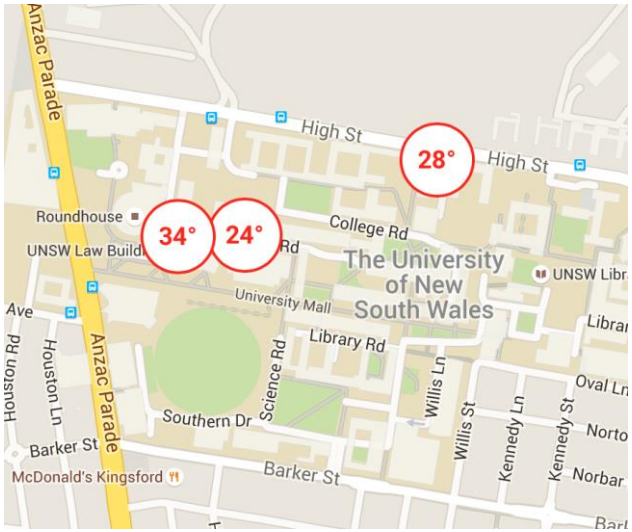
HazeWatch: Air Pollution Monitoring

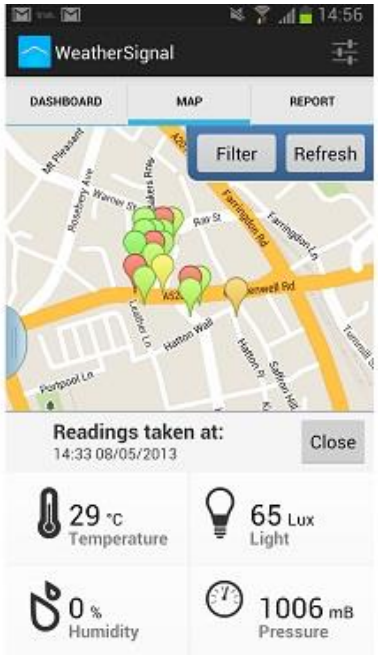




Mobile Crowdsourcing: Environment

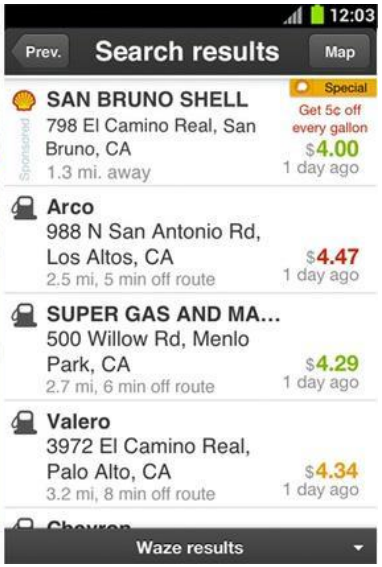
WeatherSignal: Weather Map





Mobile Crowdsourcing: Traffic

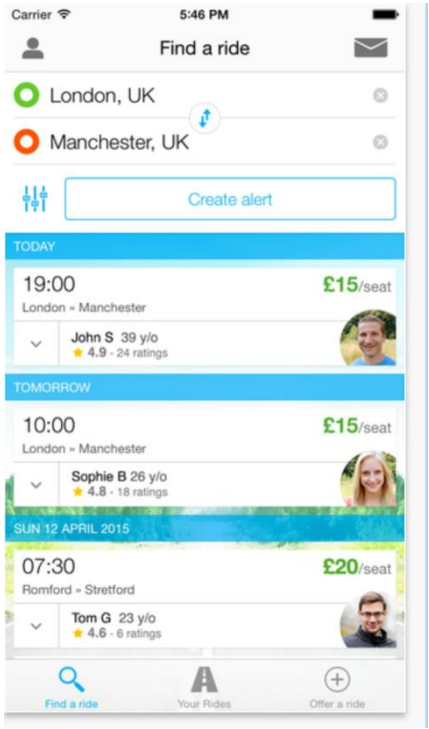
Waze: Real-time Road Information



Mobile Crowdsourcing: Transport

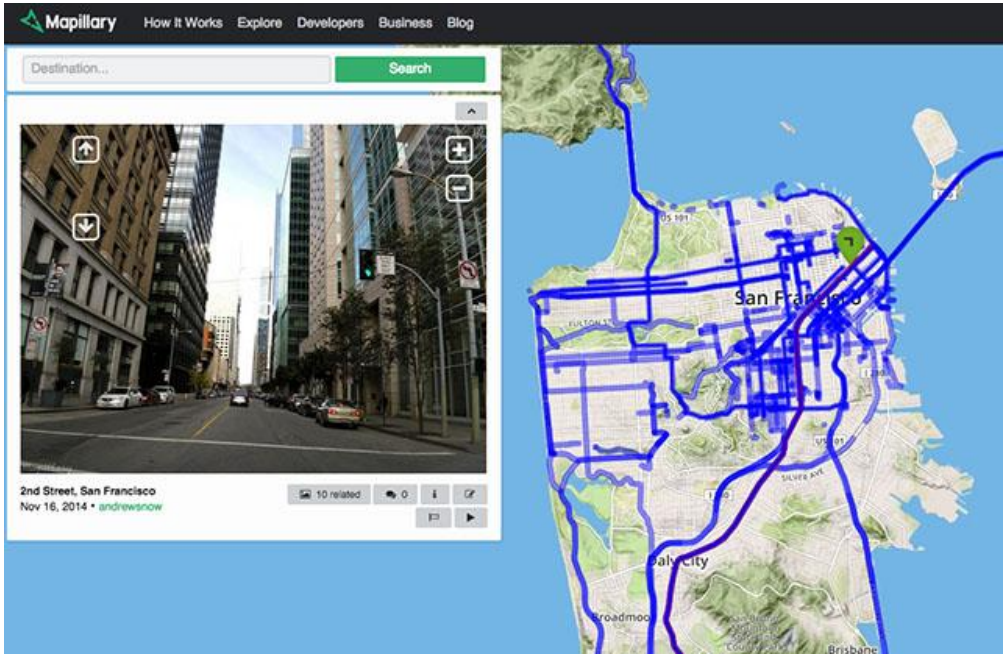
Moovit: Real-time Public Transport Information

BlaBaCar: Ridesharing

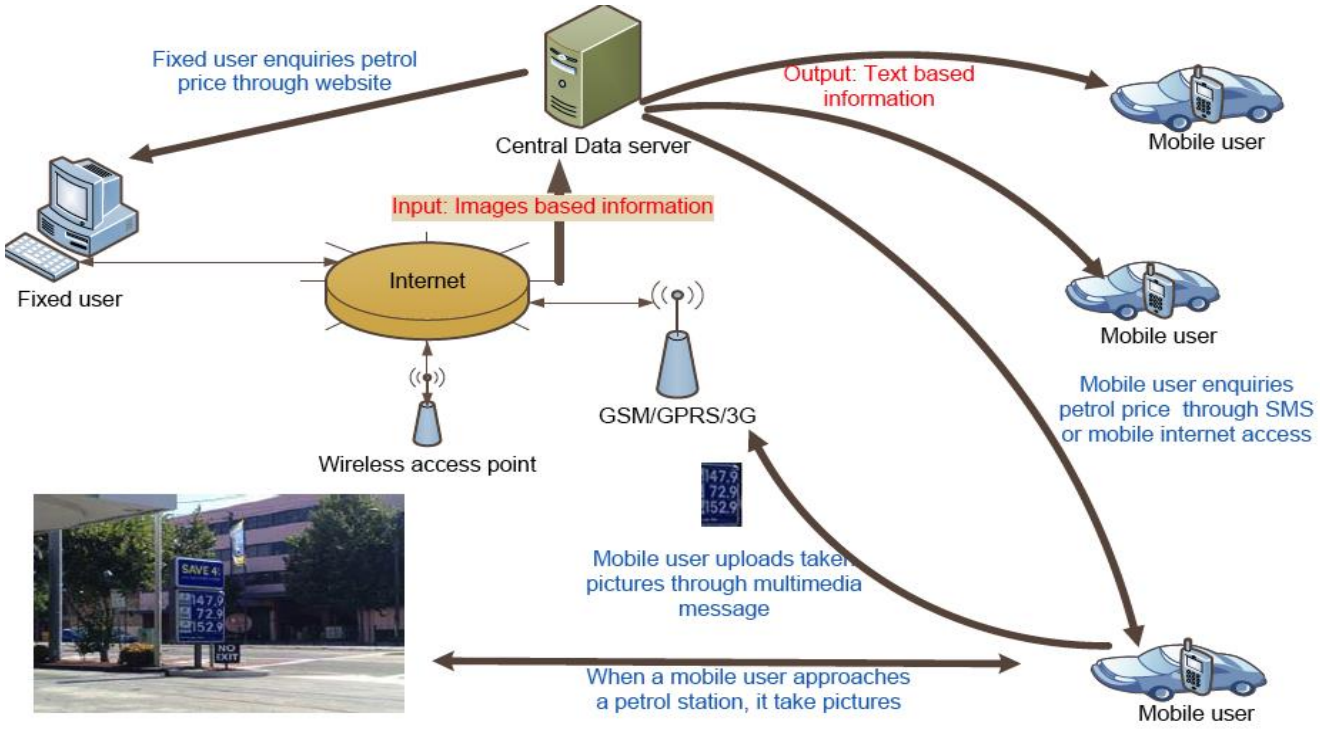


Mobile Crowdsourcing: Photos

Mapillary: Crowdsourcing Geo-tagged Photos



Mobile Crowdsourcing: Price Dispersion



PetrolWatch: Sharing Fuel Prices

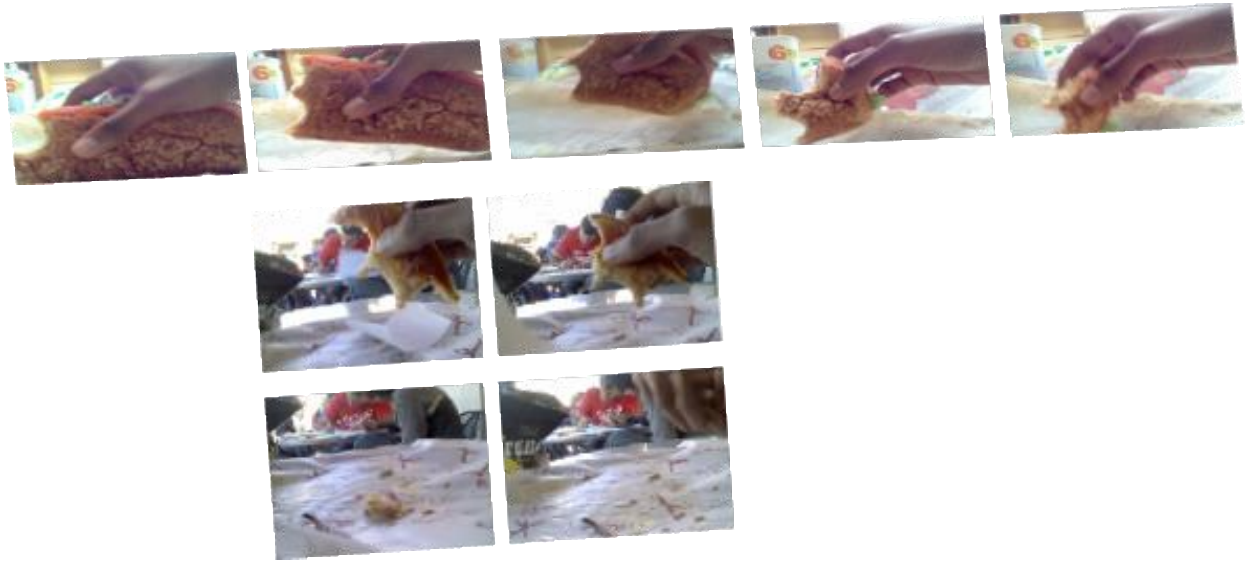
Y. Dong, S. S. Kanhere, C. T. Chou and N. Bulusu, "Automatic Collection of Fuel Prices from a Network of Mobile Cameras", in Proceedings of IEEE DCOSS 2008, Santorini, Greece, June 2008.

Mobile Crowdsourcing: Diet

mobile phone acting as sensor, worn on lanyard, automatically collecting time-stamped images of food choices/ purchases. Augmented with voice annotation, location stamping, text message alerting

DietSense: Dietary data collection via mobile crowdsourcing

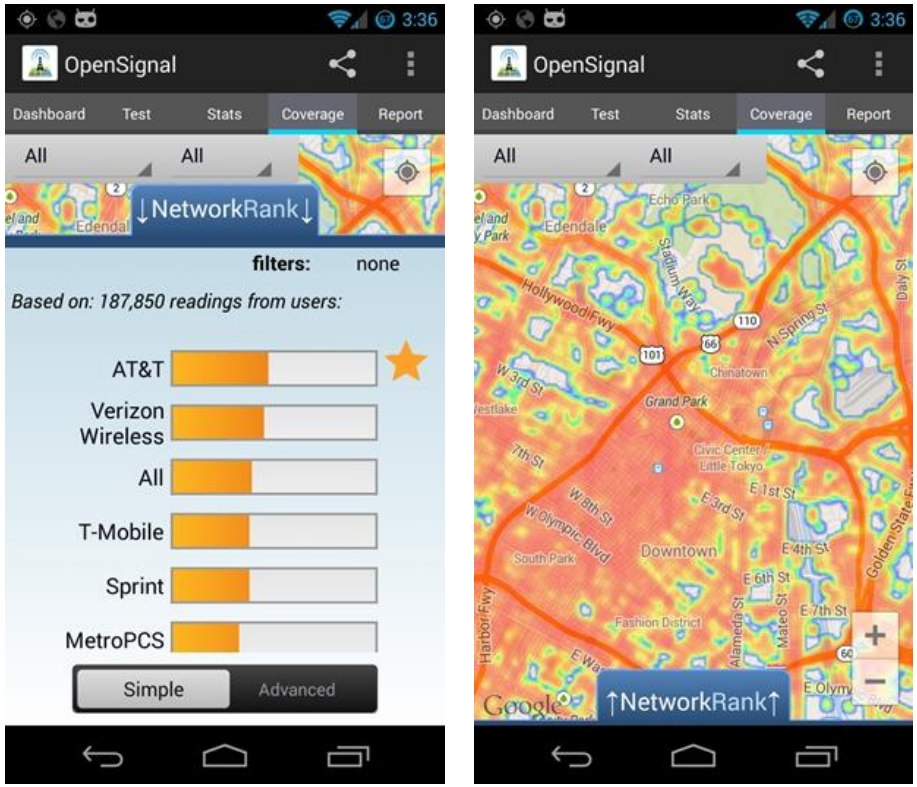
- End users initiate autonomous data capture and upload on (worn) mobile devices
- Just-in-time annotation and privacy filtering
- Tools to assist participants and dietitians in



S. Reddy, A. Parker, J. Hyman, J. Burke, D. Estrin and M. Hansen, “Image Browsing, Processing, Clustering for Participatory Sensing: Lessons from a DietSense Prototype”, in Proceedings of ACM EmNeTs, Cork, Ireland, June 2007.

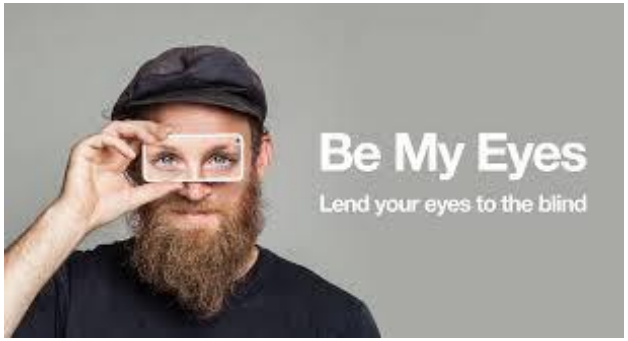
Mobile Crowdsourcing: Connectivity

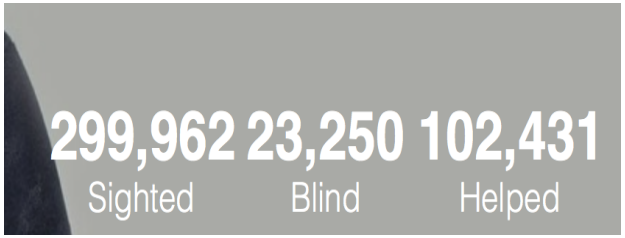
OpenSignal: Wireless Coverage Mapping

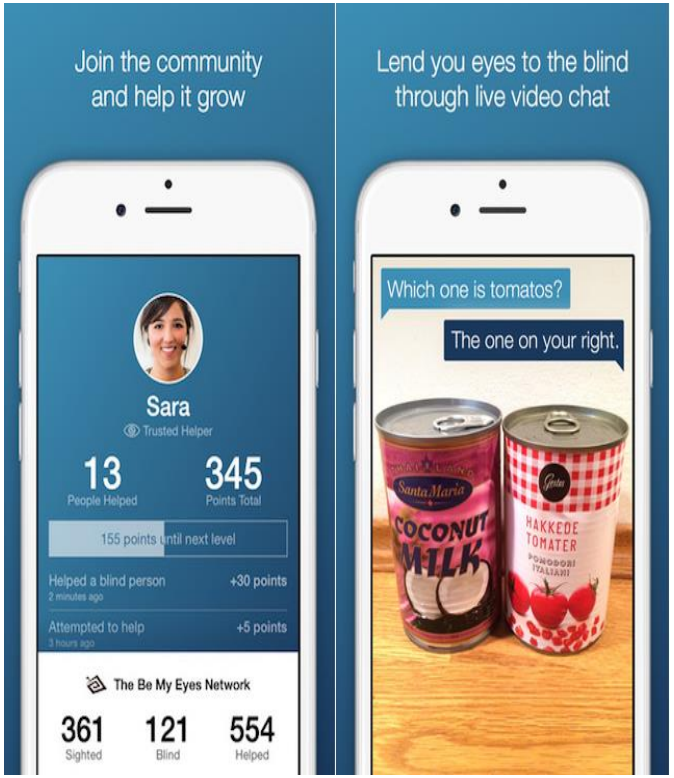


FireChat: Crowdsourcing Connections

Mobile Crowdsourcing: Volunteering

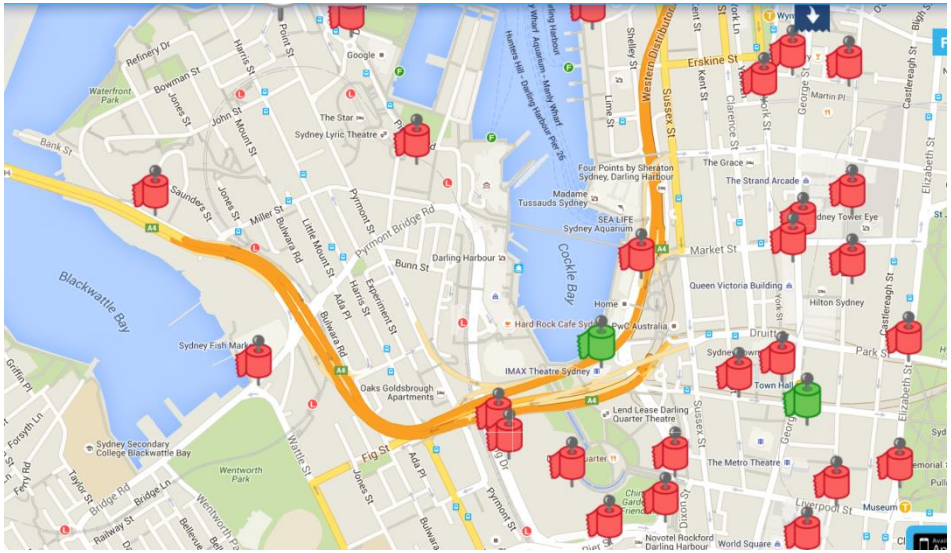






Mobile Crowdsourcing: Citizen Sensing

Sit Or Squat: Restroom Finder



Participation rates are often very low

How do we motivate people to contribute?

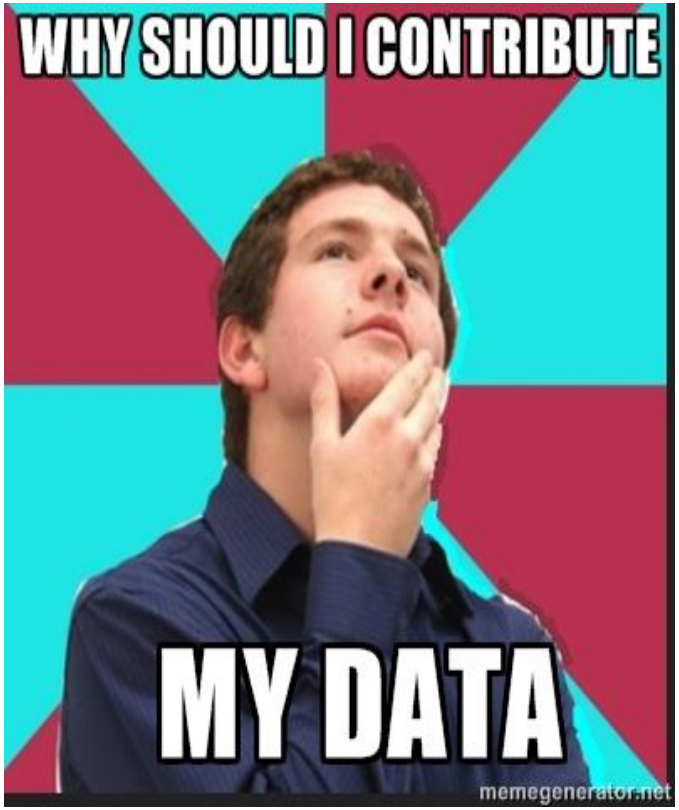
Significant costs involved:

•Time and Effort

•Phone Resources: CPU, Memory, Power, ..

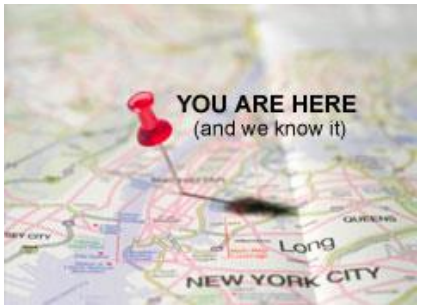
•Bandwidth

•Privacy



Q: What do I gain?

Privacy Issues



Location

+

Time

+

Activity

+

Sensor Readings

+

Background information

=

YOU !!

New Posts
+2 posts this hour

Most Popular
Most Liked NFL Players

Lists
Best Small Companies

How Target Figured Out A Teen Girl Was Pregnant Before Her Father Did

296 comments, 164 called-out

+ Comment now

Every time you go shopping, you share intimate details about your consumption patterns with retailers. And many of those retailers are studying those details to figure out what you like, what you need, and which coupons are most likely to make you happy. Target, for example, has figured out how to data-mine its way into your womb, to figure out whether you have a baby on the way long before you need to start buying diapers.

Target has got you in its aim

Charles Duhigg outlines in the New York Times how Target tries to hook parents-to-be at that crucial moment before they turn into rampant — and loyal — buyers of all things pastel, plastic, and miniature. He talked to Target statistician Andrew Pole — before Target freaked out and cut off all communications — about the clues to a customer's impending bundle of joy. Target assigns every customer a Guest ID number, tied to their credit card, name, or email address that becomes a bucket that stores a history of everything they've bought and any demographic information Target has collected from them or bought from other sources. Using that, Pole looked at historical buying data for all the ladies who had signed up for Target baby registries in the past. From the NYT:

Source: Forbes Magazine

Data Trustworthiness

Inherent openness of the urban sensing paradigm means anyone can contribute data

Users may inadvertently contribute low quality data

Malicious users may knowingly contribute false data

Q: How can we trust the data received?

PART I.
INCENTIVES

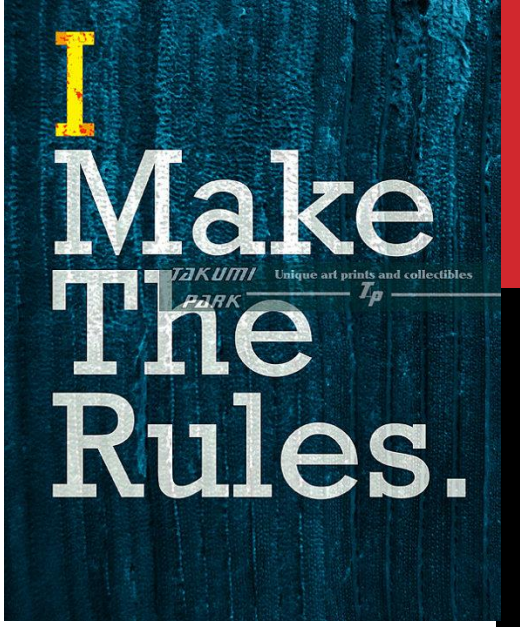
Roadmap

- Introduction

- **Incentives**
  - **Fundamentals of mechanism design**
    - Bayesian mechanism design
  - Crowdsourcing and all-pay auctions
  - Tullock contests

- Trust
- Privacy

Fundamentals of Mechanism Design



"**Reverse game theory**"

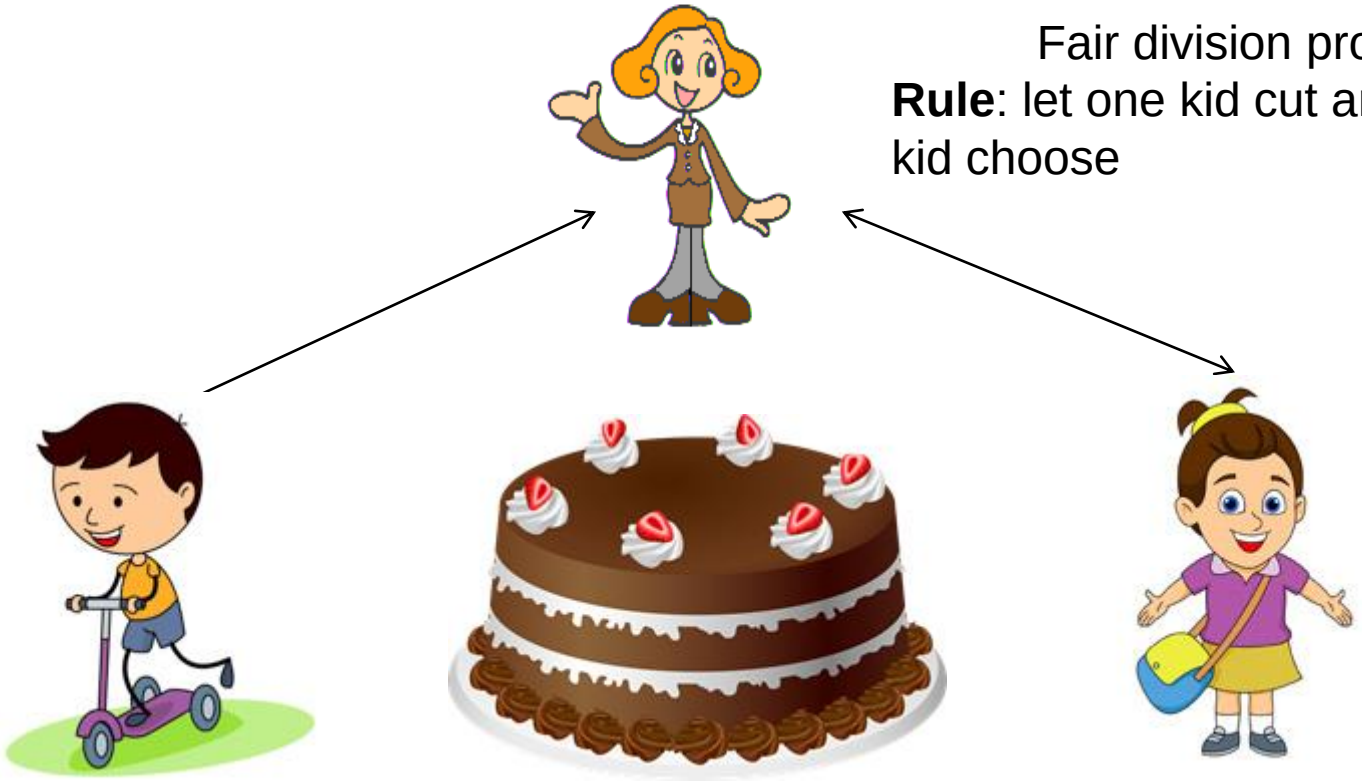
- GT: given the game rules, you reason about how all the players behave
- MD: you (as the designer) specify rules such that players behave as you desire

Overarching goal

Design a mechanism such that, despite players are **strategic** and may have **different interest** from designer's, the system functions "well"

Example 1

Cake Cutting: How to ensure fairness?



Fair division problem

Rule: let one kid cut and the other kid choose

Example 2

How to ensure the item goes to the person who wants it the most?

1 150

2 180

3  200

4 120

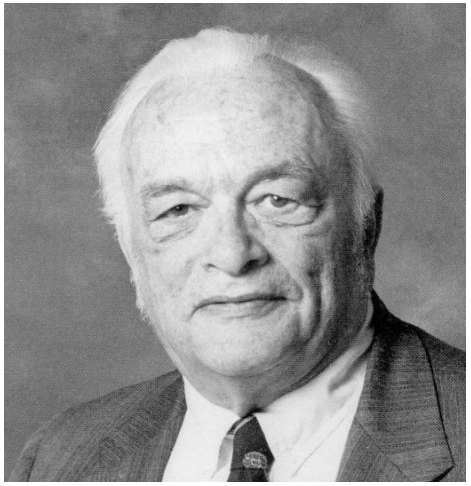
Solution 1 (naïve):
“ask” buyers

Solution 2:
“pay what you claim”

Example 2 (cont'd)

Second-price auction:

- Allocation rule: highest bidder
- Payment rule: 2nd-highest bid

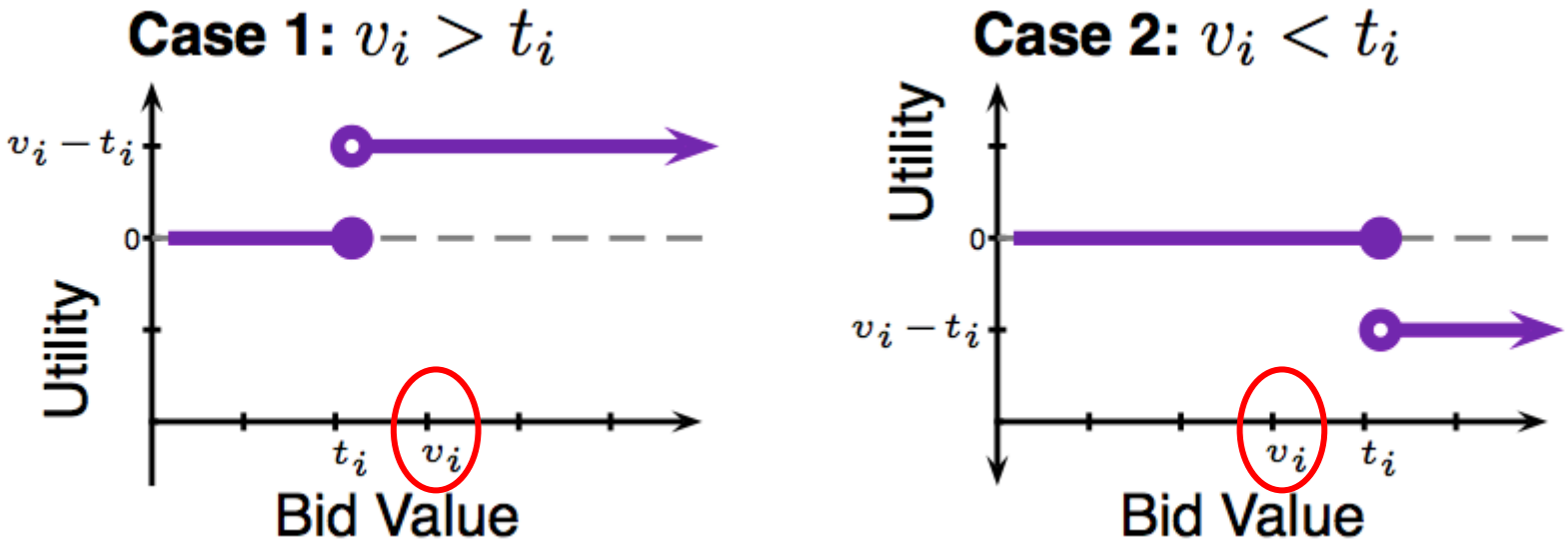


William Vickrey
(Nobel Prize 1996)

Dominant-strategy incentive-compatible (DSIC)

Vickrey auction

- Utility = value – payment
- Value: v_i
- Bid: b_i
- $t_i = \max_{j \neq i} b_j$ (the highest bid of the others)



Real application: eBay

Apple iPad 2 32GB 2nd Gen Wi-Fi 9.7in Tablet MC770LL/A - Black

Like Want Own ★★★★½ 167 product reviews

Item condition: **Used**

Time left: 23h 52m 24s (Dec 30, 2012 14:00:56 PST)

Current bid: **US $0.99** [1 bid]

Place bid

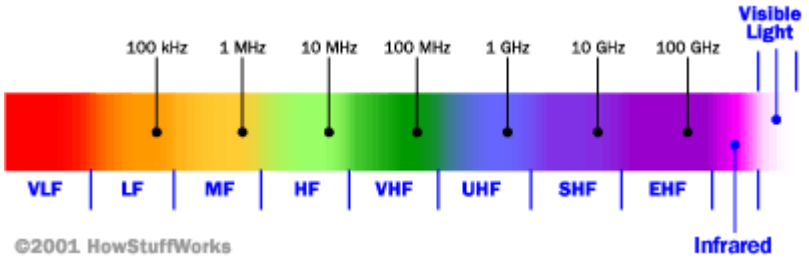
Enter US $1.04 or more

Add to Watch list

BillMeLater **New customers get $10 back on 1st purchase**
Subject to credit approval. See terms

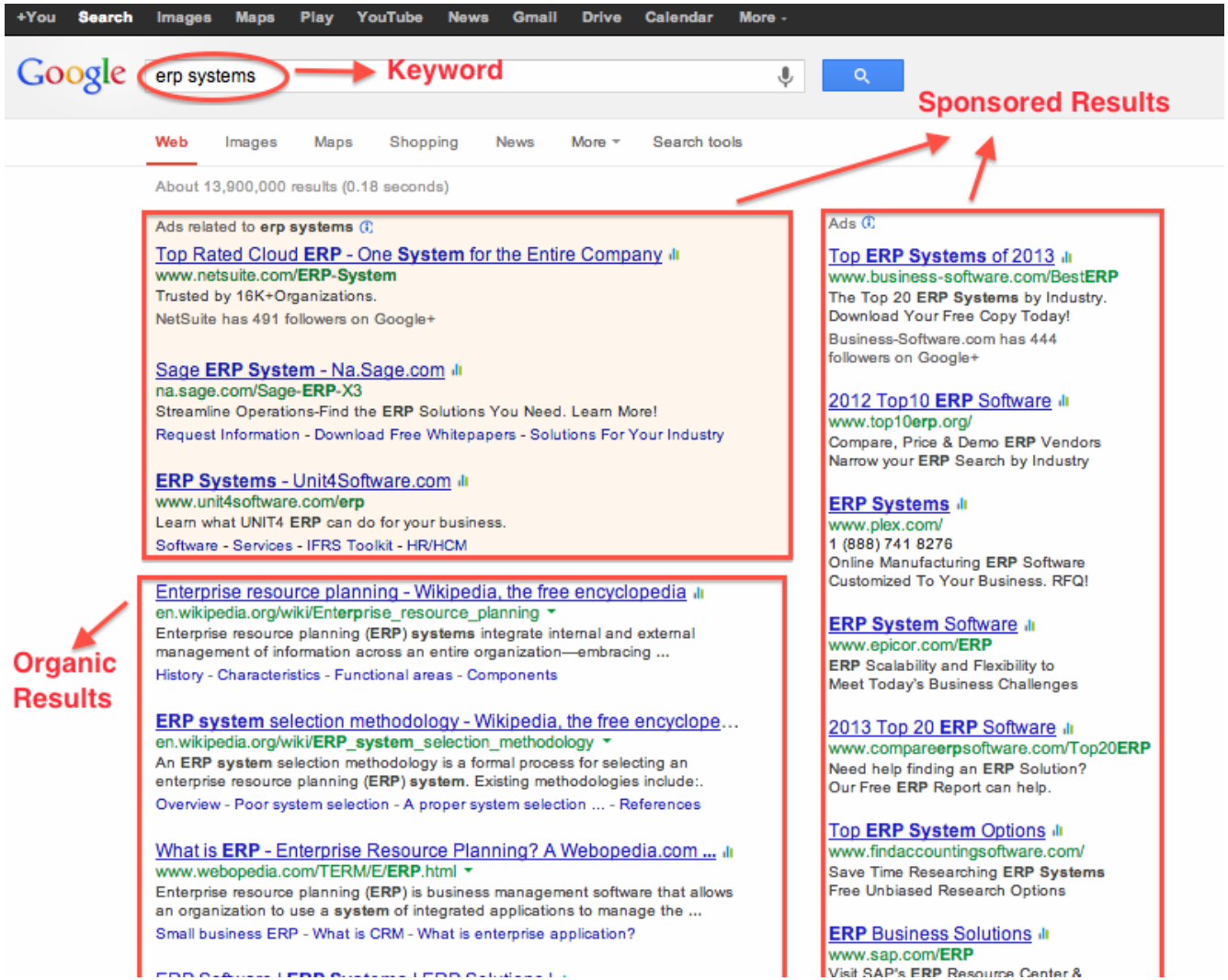
Radio spectrum auction

(simplified from *combinatorial auction*
For heterogeneous spectrum licenses)



Federal Communications Commission

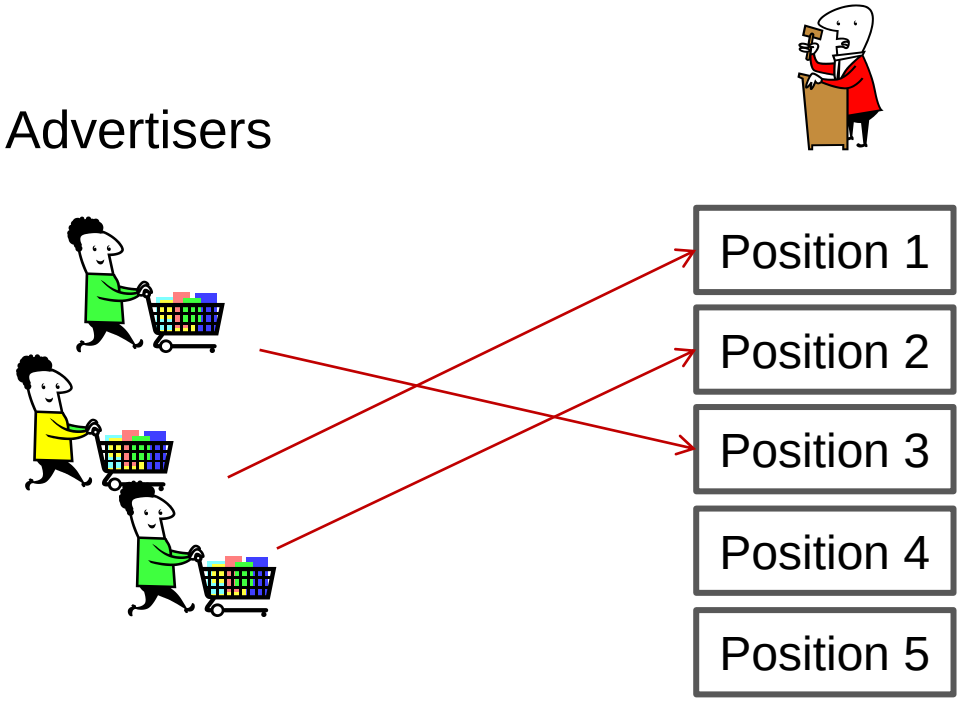
Example 3



Sponsored search:

How to place ads so that revenue of the search engine is maximized?

Sponsored search



Multi-item auction: Generalized second-price auction & Vickrey–Clarke–Groves (VCG) auction

Allocation rule: assign ad slots in order of bids (scaled by relevance)

Payment rule: the externality each bidder causes to other bidders

- Hal Varian, Christopher Harris. The VCG Auction in Theory and Practice, In The American Economic Review,
- Q. Liu, T. Luo, R. Tang, and S. Bressan , An Efficient and Truthful Pricing Mechanism for Team Formation in Crowdsourcing Markets, IEEE ICC, June 2015, pp. 567-572.

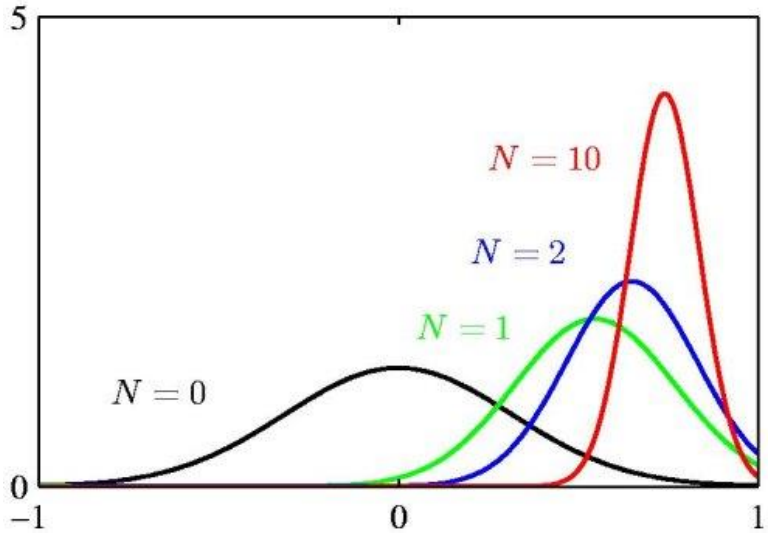
Private information ('type')

- Auction: valuation of item
- Crowdsourcing: contribution cost, desired payment, etc.

Bayesian mechanism design

Incomplete Information

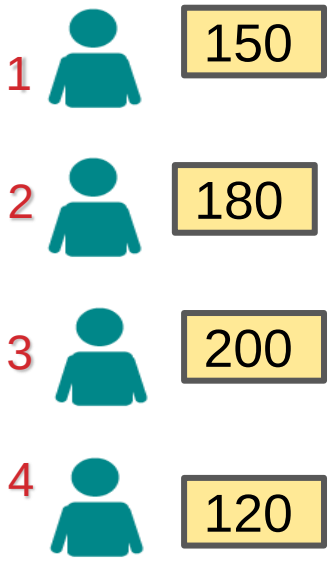
Although the exact private information is unknown, we may have some **probabilistic knowledge** about it, for example its distribution



Probabilistic knowledge may be derived from:

- Historical market data
- Domain-specific knowledge
- Presumption of natural inputs

First-price auctions



- 1 item to sell
- n potential buyers, with values $\boldsymbol{v} = v_1, v_2, \ldots, v_n$ for the item
- $\boldsymbol{v}$ is private information (unknown)
- **Common prior assumption**: $\boldsymbol{v} \sim \boldsymbol{F} = F_1 \text{ x } F_2 \text{ x } \ldots F_n$ (known)
- Buyer utility: $u_i = v_i$ - payment (if win) or 0 (if lose)
- Bidding strategy $b_i(v_i)$
- **First price: payment = bid**

Bayes-Nash equilibrium

A strategy profile $\boldsymbol{b}^* = (b_1^*, b_2^*, \ldots, b_n^*)$ is a **Bayes-Nash equilibrium** if, for each i and v_i, $b_i^*(v_i)$ maximizes player i's expected utility u_i, given that others play b_{-i}^*. That is,

$$E_{\boldsymbol{v}\sim\boldsymbol{F}}[u_i(b_i^*, b_{-i}^*)|v_i] \geq E_{\boldsymbol{v}\sim\boldsymbol{F}}[u_i(b_i, b_{-i}^*)|v_i], \qquad \forall\, i, \forall\, v_i$$

Equilibrium analysis

Player's goal: maximize **expected utility**

$$
\begin{aligned}
E[u_i] &= \Pr[i \text{ wins}](v_i - b_i) + \Pr[i \text{ loses}] \bullet 0 \\
&= \Pr[b_i > b_j, \forall j \neq i](v_i - b_i) \\
&= \prod_{j \neq i} F_j(v_i) \times (v_i - b_i) \quad \text{//assuming monotone increasing bids} \\
&= F^{n-1}(v_i) \times (v_i - b_i) \quad \text{//assuming i.i.d. (symmetric case)} \\
&= v_i^{n-1} \times (v_i - b_i) \quad \text{//assuming uniform distribution } [0,1]
\end{aligned}
$$

First-order condition w.r.t. b_i:

$$\frac{\mathrm{d}}{\mathrm{d}b} v^{n-1}(v_i - b) = \frac{(n-1)v^{n-2}}{b'}(v_i - b) - v^{n-1}$$

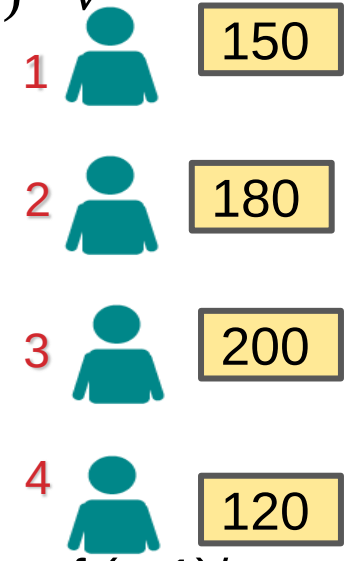
Equating it to zero, we have

$$b' = (n-1)(1-\frac{b}{v})$$

which leads to*

$$b = \frac{n-1}{n} v$$

¾ x 200=150



So, each bidder **shades down** his bid by a factor of (n-1)/n.

Special cases: n=2, b=v/2; n=3, b=2v/3

* To solve the (linear) differential equation, assume b = c1 v + c2

Revenue

Since $$E[v_{(n)}] = \int_0^1 v g(v) dv = vG(v)\,|_0^1 - \int_0^1 G(v) dv = 1 - \int_0^1 G(v) dv$$

$$\because G(v) = F^n(v) = v^n$$

$$\therefore E[v_{(n)}] = 1 - \frac{v^{n+1}}{n+1}\,|_0^1 = \frac{n}{n+1}$$

Therefore, the revenue is

$$E[b_{(n)}] = \frac{n-1}{n} E[v_{(n)}] = \frac{n-1}{n+1}$$

Second-price auctions

In the second-price case: $b = v$

Revenue is: $E[v_{(n-1)}] = \frac{n-1}{n+1}$

the same as first-price auction.

Revenue equivalence theorem (RET)

Consider an auction in which each of the *n* risk-neutral bidders has a privately known value drawn independently from a common, strictly increasing distribution. If

- the item goes to the bidder with the highest value,
- any bidder with the lowest value expects zero utility,

then any such auction yields the **same expected revenue**.

[Vickrey (1961), Myerson (1981), Riley and Samuelson (1981), Harris and Raviv (1981)]

Roadmap

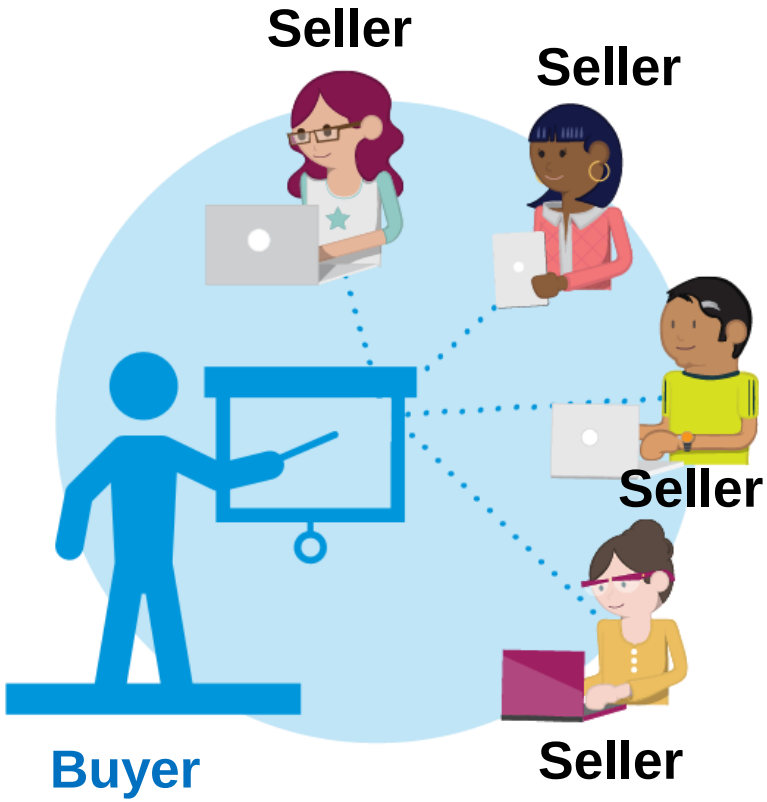
- Introduction
- Incentives
  - Fundamentals of mechanism design
    - Bayesian mechanism design
  - **Crowdsourcing and all-pay auctions**
  - Tullock contests
- Trust
- Privacy

Traditional auction:

One seller, multiple buyers

Crowdsourcing - Reverse auction:

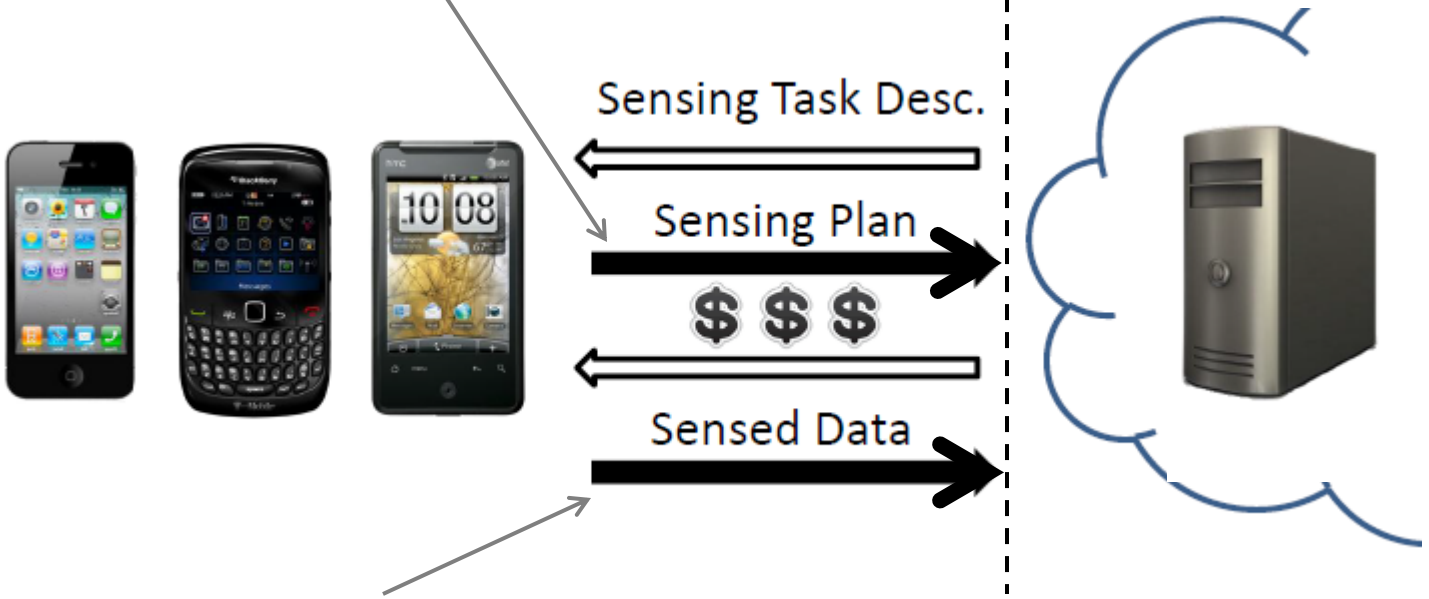
- One buyer: crowdsourcer / requester
- Multiple sellers: workers / contributors, e.g., sell mobile sensing data



Procedures of reverse auction

1. **Bidding stage:**
   users submit bids (e.g., desired reward)

2. Winner selection (e.g., those who ask for a lower reward)



3. Contribution stage: winners contribute and receive reward

[Lee & Hoh'10, Yang et al.'12, Koutsopoulos'13, Feng et al.'14, Zhao et al.'14, …]

State of the art: Winner-pay auctions

Only the winner(s) pay for the item

- Example: first-price and second-price auctions

Most crowdsourcing incentive mechanisms belong to this category:

- Only selected users perform task (contribute and receive reward)

J.-S. Lee and B. Hoh, “Sell your experiences: A market mechanism based incentive for participatory sensing,” in IEEE PerCom, 2010.

D. Yang, G. Xue, X. Fang, and J. Tang, “Crowdsourcing to smartphones: Incentive mechanism design for mobile phone sensing,” in ACM MobiCom, 2012.

I. Koutsopoulos, “Optimal incentive-driven design of participatory sensing systems,” in IEEE INFOCOM, 2013.

Z. Feng, Y. Zhu, Q. Zhang, L. Ni, and A. V. Vasilakos, “TRAC: Truthful auction for location-aware collaborative sensing in mobile crowdsourcing,” in IEEE INFOCOM, 2014.

D. Zhao, X.-Y. Li, and H. Ma, “How to crowdsource tasks truthfully without sacrificing utility: Online incentive mechanisms with budget constraint,” in IEEE INFOCOM, 2014.

All-pay auctions

Everyone pays his bid, regardless of who wins.

In hindsight – a natural fit for crowdsourcing:

Bid = Contribution

- Once contribution is submitted, user effort is sunk and irrevocable
- Essentially, all bids are "*paid*" once submitted

Comparison with winner-pay

Winner-pay auctions:

1. **Bidding stage**: bids indicate users' willingness to contribute
2. Select users based on bids ("*promises*")
3. **Contribution stage**

Risk of non-fulfillment

(intentionally or unintentionally)

Promise ➔ Actual contribution

All-pay auctions

Three advantages:

- **Simplicity**: compresses the two-stage "bid-and-contribute" process into a single "**bid-cum-contribute**" stage
- **Risk free**: eliminates risk of **task non-fulfillment**
- Inherently **truthful** (incentive compatible): winner selection is based on actual (and observable) contribution which internalizes user's (private) *type* (ability/cost) and cannot be lied about [TIST'16, page 8]

T. Luo, S. K. Das, H-P. Tan, and L. Xia, "Incentive mechanism design for crowdsourcing: an all-pay auction approach", ACM TIST, vol. 7, no. 3, pp. 35:1-26, 2016.

Equilibrium analysis

Bidding strategy & **revenue**

Utility: $u = v - b$ (if win) or $-b$ (if lose)

Expected utility: $\mathrm{E}[u]$

$$= \Pr[i \text{ wins}]v_i - b_i$$

$$= \Pr[b_i > b_j, \forall j \neq i]v_i - b_i$$

$$= \prod_{j \neq i} F_j(v_i) \times v_i - b_i \quad \text{//assuming monotone increasing bids}$$

$$= F^{n-1}(v_i) \times v_i - b_i \quad \text{//assuming i.i.d.}$$

$$= v_i^{n-1} v_i - b_i \quad \text{//assuming uniform between } [0,1]$$

First-order condition w.r.t. b:

$$\frac{\mathrm{d}}{\mathrm{d}b}(v^{n-1}v_i - b) = \frac{(n-1)v^{n-2}}{b'}v_i - 1$$

Equating it to zero, we have

$$b' = (n-1)v^{n-1}$$

which leads to*

$$b = \frac{n-1}{n}v^n$$

Compare to first price auction: $b = \frac{n-1}{n}v$

each bidder shades down bid significantly!

* To solve the (linear) differential equation, assume b = c1 v + c2

Revenue

$$E[\sum_i b_i] = \frac{n-1}{n}\sum_i E[v^n] = \frac{n-1}{n}\sum_{i=1}^{n}\int_0^1 v^n dv = \frac{n-1}{n+1}$$

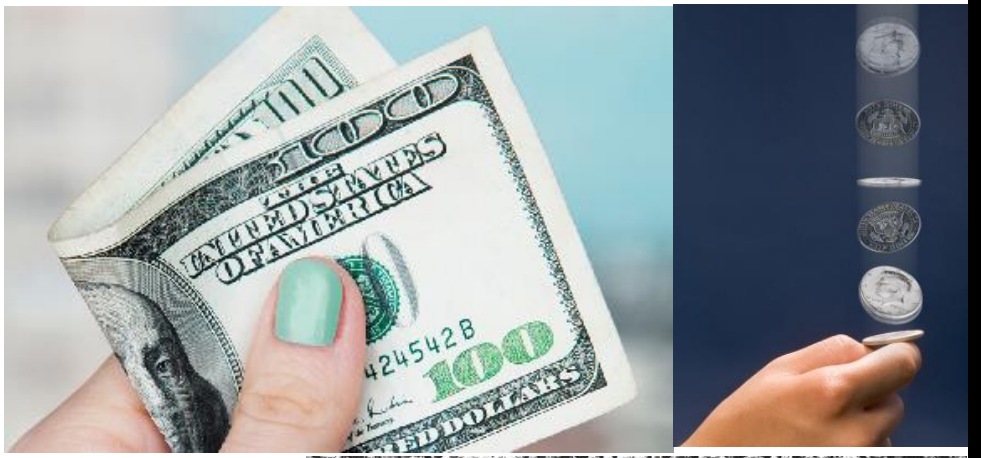
Revenue equivalence

On the one hand, significant bid-shading

On the other hand, revenue is composed of *n* bids

Put auctions into practice?

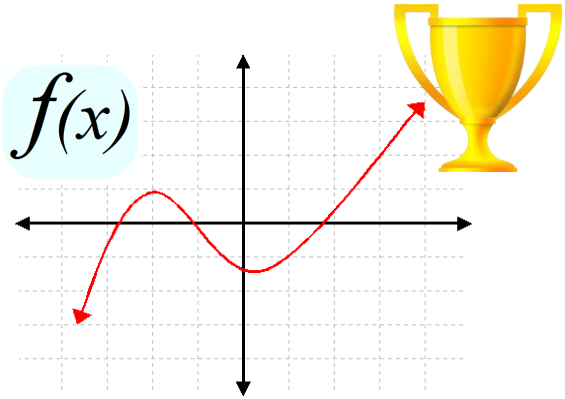
1. Non-standard settings:
   - Risk-averse players
   - Stochastic population



Perturbation theory

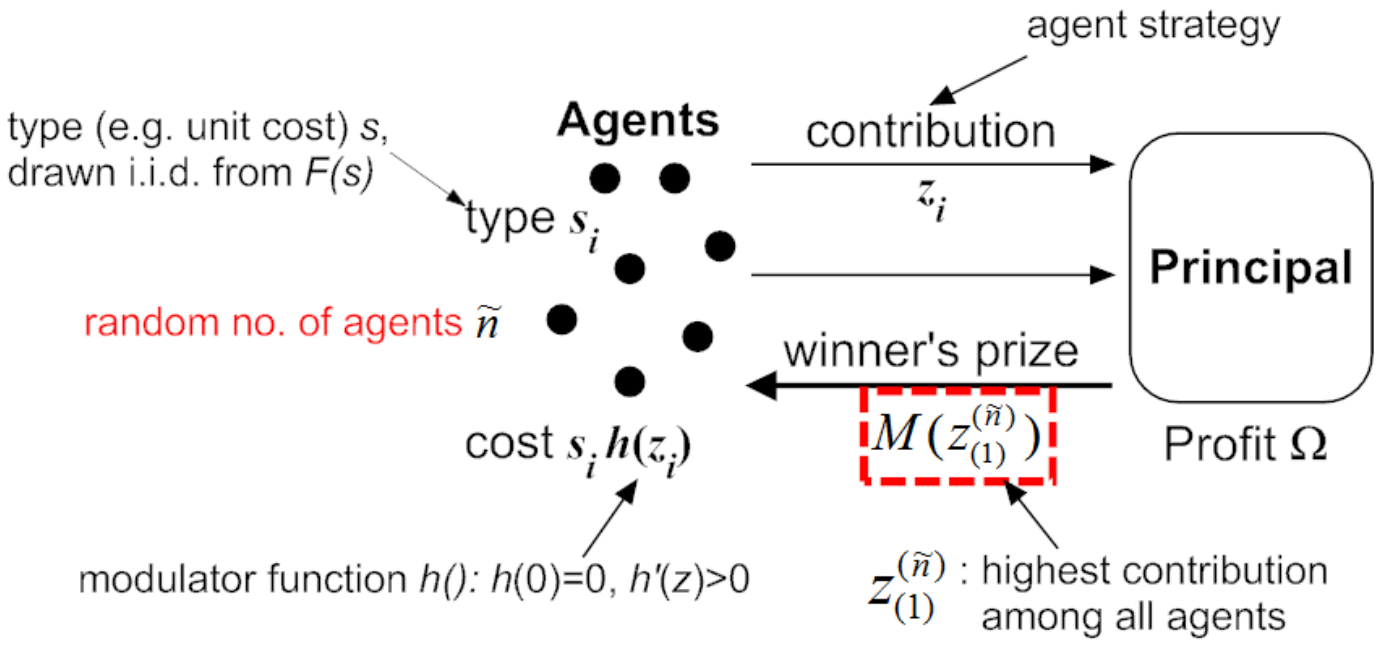
2. Higher revenue notwithstanding RET?

Adaptive prize



T. Luo, H-P. Tan, and L. Xia, “Profit-Maximizing Incentive for participatory sensing”, IEEE INFOCOM, 2014.

Model



von Neumann-Morgenstein utility function

Agent's utility: $$\tilde{\pi}_i(s_i, \boldsymbol{z}) = \begin{cases} u(M(z_i) - s_i h(z_i)), & \text{if } z_i > z_j, \forall j \neq i; \\ u(-s_i h(z_i)), & \text{otherwise,} \end{cases}$$

Principal's profit: $$\Omega(\tilde{n}, \boldsymbol{z}) := \sum_{i=1}^{\tilde{n}} z_i - M(z_{(1)}^{(\tilde{n})})$$

Main results

Optimal prize function: $$\mathring{M}(z) = \frac{1}{P(\mathring{s}(z))}\left[\mathring{s}(z)h(z) - \mathring{A}(\mathring{s}(z)) - \int_{\mathring{z}(\underline{s})}^{z} \mathring{B}(\mathring{s}(z_1))h(z_1)\,\mathrm{d}\mathring{s}(z_1)\right]$$

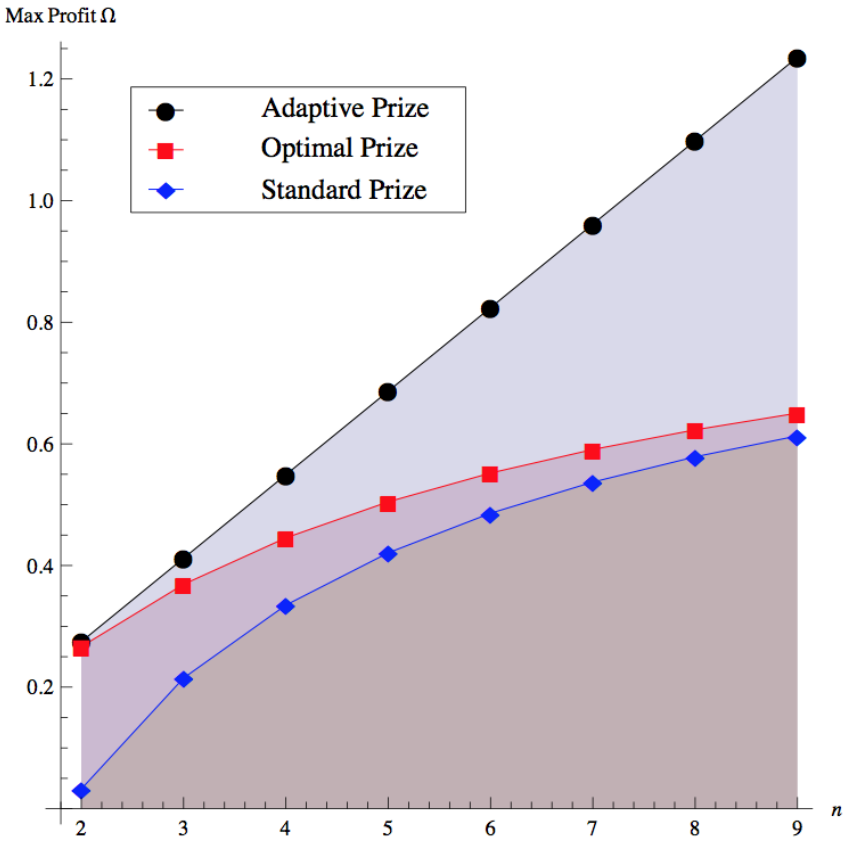
Agent's optimal strategy: $$\mathring{z}(s) = (h')^{-1}\left(\frac{aF'(s)}{G'(s)s + G(s)\mathring{B}(s)}\right)$$

Principal's max profit: $$\mathring{\Omega} = \int_{\underline{s}}^{\bar{s}} \left[a\mathring{z}F' - h(\mathring{z})\left(sG' + \mathring{B}(s)G\right) + \mathring{A}(s)G'\right]\mathrm{d}s.$$

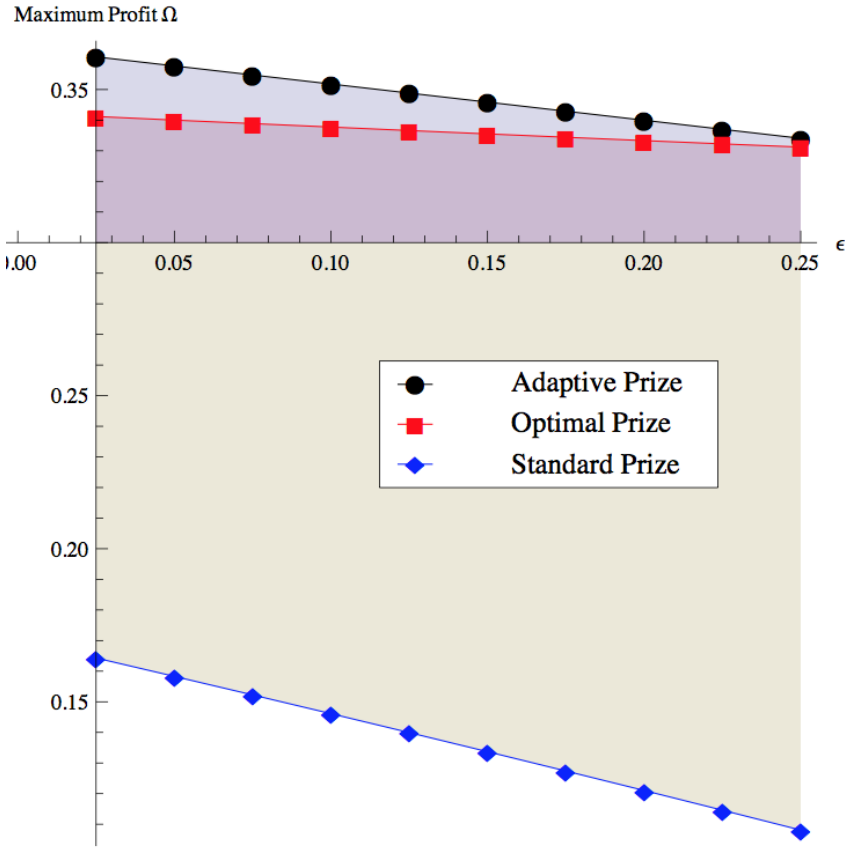
Strict individual rationality (SIR): agents strictly have incentive to participate

Profit comparison

- **Standard prize**: normalized prize = 1 (fixed prize)
- **Optimal prize**: optimized s.t. profit is maximized (fixed prize)
- **Adaptive prize**: our incentive mechanism



Profit vs. no. of players



Profit vs. risk aversion

Recap

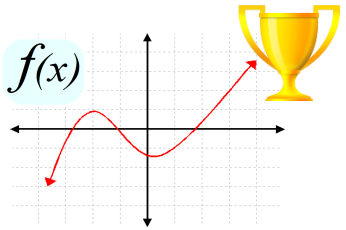
All-pay auctions' three merits:

- Simplicity: "2 in 1"
- Risk free
- Inherently truthful

General problem setting:

- Incomplete information (Bayesian)
- Risk averse
- Stochastic population

Adaptive prize for revenue maximization



T. Luo, S. K. Das, H-P. Tan, and L. Xia, "Incentive mechanism design for crowdsourcing: an all-pay auction approach", ACM TIST, vol. 7, no. 3, pp. 35:1-26, 2016.

Outline

- Introduction

- **Incentives**
  - Fundamentals of mechanism design
    - Bayesian mechanism design
  - All-pay auctions
    - **Heterogeneous all-pay auctions**
  - Tullock contests

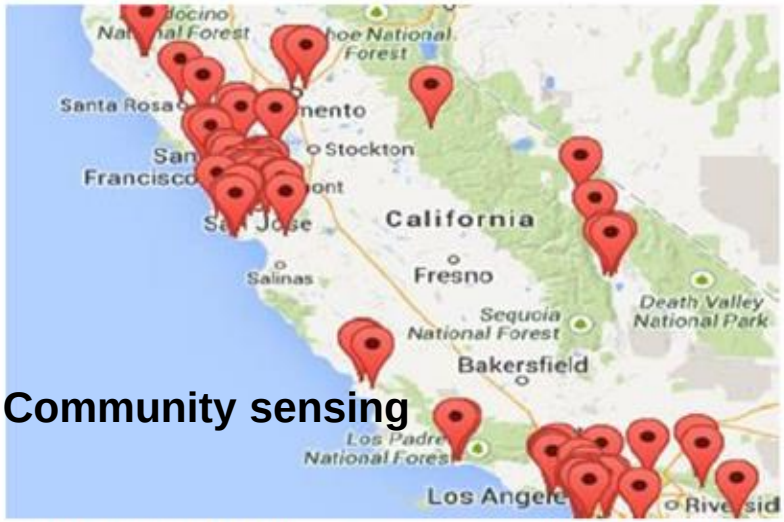
- Trust
- Privacy

More realistic scenarios

Previously we have assumed a single common prior:

- All players' private information (type) follows the same probability distribution F (most common in the literature)
  - e.g., contribution cost of every agent ~ U[0,1] and i.i.d.

Reality may deviate from this homogeneous setting



Community sensing

HIGHEST CONTRIBUTION POWER		HIGHEST ENDORSEMENT POWER	
Rank	Users	CP Score	EP Score
1	Rookie Ac	392.23	115.15
2	Jason Pan	179.41	143.80
3	Ting Ting	163.52	123.12
4	Zhang Xinwan	150.32	100.00
5	Jasmine	148.32	114.52

Publicized performance

(a) Creek Watch [2]: Waterway conditions reported by iPhone users. (b) WiFi-Scout [20]: user ranking based on contribution performance.

Heterogeneous players

Each player's private information follows a different distribution: $F_1, F_2, \ldots, F_n$

Asymmetric auction

Can adaptive prize apply to this heterogeneous setting?

- Challenge: solving asymmetric auctions even for fixed prizes is an **open problem** in general
  - analytical solutions are only available for special cases such as **two players** or **complete information**
- Yet n players with incomplete information is generally more useful

Key idea

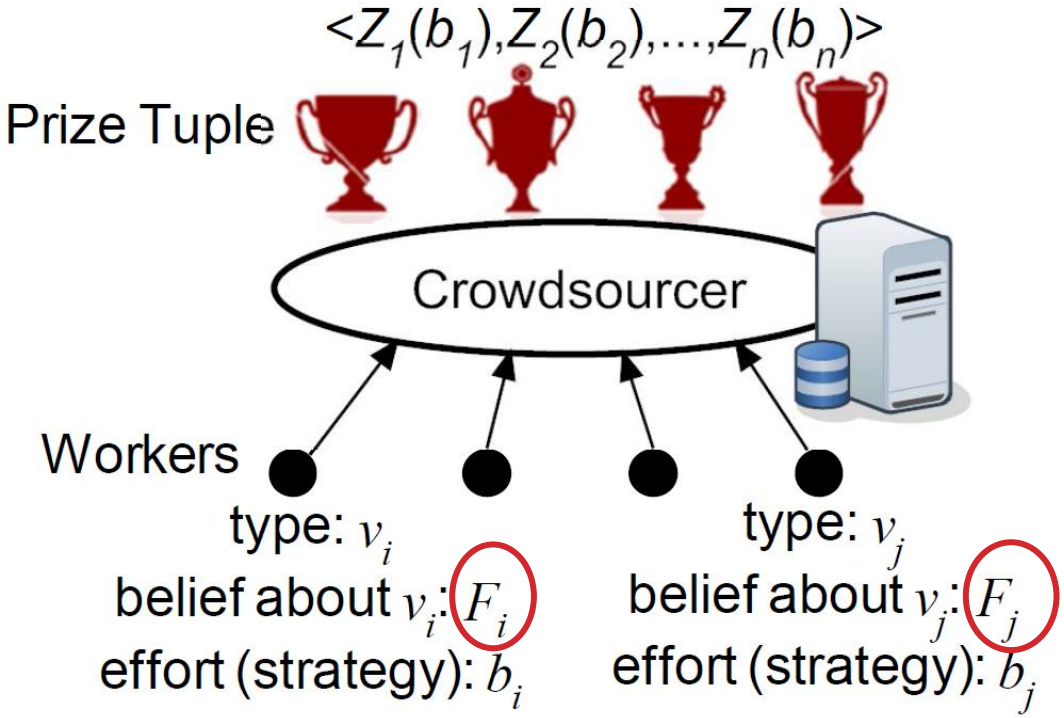
Introduce **prize tuple**

- an array of adaptive prizes

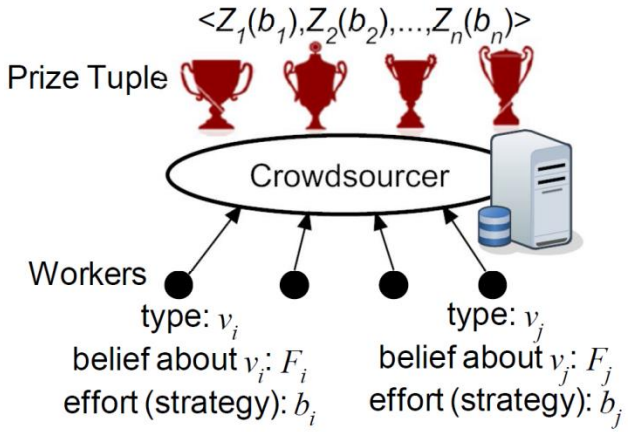
$$\langle Z_1(b_1), Z_2(b_2), \ldots, Z_n(b_n) \rangle$$

T. Luo, S. S. Kanhere, S. Das, and H-P. Tan, “Optimal Prizes for All-Pay Contests in Heterogeneous Crowdsourcing”, IEEE MASS, 2014.

Model


<Z1(b1),Z2(b2),...,Zn(bn)>
Prize Tuple
Crowdsourcer
Workers
type: v_i
belief about v_i: F_i
effort (strategy): b_i
type: v_j
belief about v_j: F_j
effort (strategy): b_j

Model (cont'd)



Worker:

- maximize $u_i = q_i\, V(v_i, Z_i) - p(b_i, v_i)$

p: cost function

q_i: winning probability

$V(v_i, Z_i)$: value of prize, e.g. $V(v_i, Z_i) = v_i Z_i$

Crowdsourcer:

- maximize $\pi = \Sigma^n_{i=1} b_i - V(\lambda, Z_w)$

b_i: effort

w: winner's index

λ: crowdsourcer's type (marginal valuation of prize)

Main result

Optimal prize tuple: $$Z_i(b_i) = \frac{\hat{p}(b_i, v_i(b_i)) - \int_0^{b_i} \hat{p}'_{v_i}(\tilde{b}_i, v_i(\tilde{b}_i))\, \mathrm{d}v_i(\tilde{b}_i)}{\prod_{j \neq i} F_j(v_j(b_i))}, \quad i = 1, 2, ..., n,$$

Optimal agent effort b_i: $$\hat{p}'_{b_i}(b_i, v_i) = \frac{1}{h(\lambda)} + \hat{p}''_{b_i, v_i}(b_i, v_i)\frac{1 - F_i}{f_i}$$ **(SA)**

Maximum profit: $$\pi = \sum_i \int_{\underline{v}}^{\bar{v}} \left[b_i(v_i) - h(\lambda)\hat{p}(b_i, v_i) + h(\lambda)\hat{p}'_{v_i}(b_i, v_i)\frac{1 - F_i}{f_i} \right] \mathrm{d}F_i$$

New property discovered: Strategy Autonomy

Each agent's equilibrium strategy is **independent** of his knowledge about other agents ($F_1, F_2, \ldots, F_n$)

- In other words, agents behave **autonomously** as if the **asymmetric** auction admits a **symmetric** equilibrium

In contrast to all classic asymmetric auctions

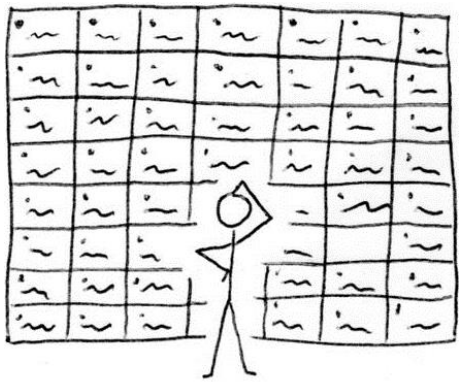
Practical implications of SA

1. **Reduces** mechanism **complexity** & energy consumption
   - from **O(n)** to **O(1)** for *each* agent

Classic auctions:

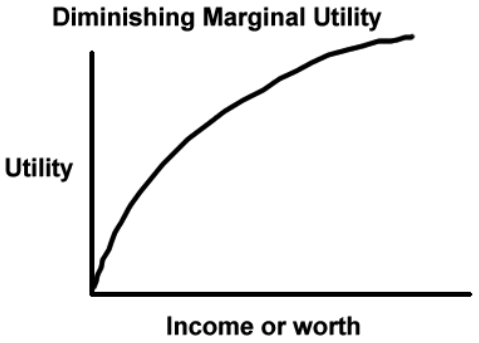
$$V(v_i, Z_i(b_i)) \prod_{j \neq i} F_j(v_j(b_i)) - p(b_i, v_i)$$

$$= \int_{\underline{v}}^{v_i} [V'_{v_i}(\tilde{v}_i, Z_i(b_i)) \prod_{j \neq i} F_j(v_j(b_i)) - p'_{v_i}(b_i, \tilde{v}_i)] \, \mathrm{d}\tilde{v}_i. \quad (4)$$

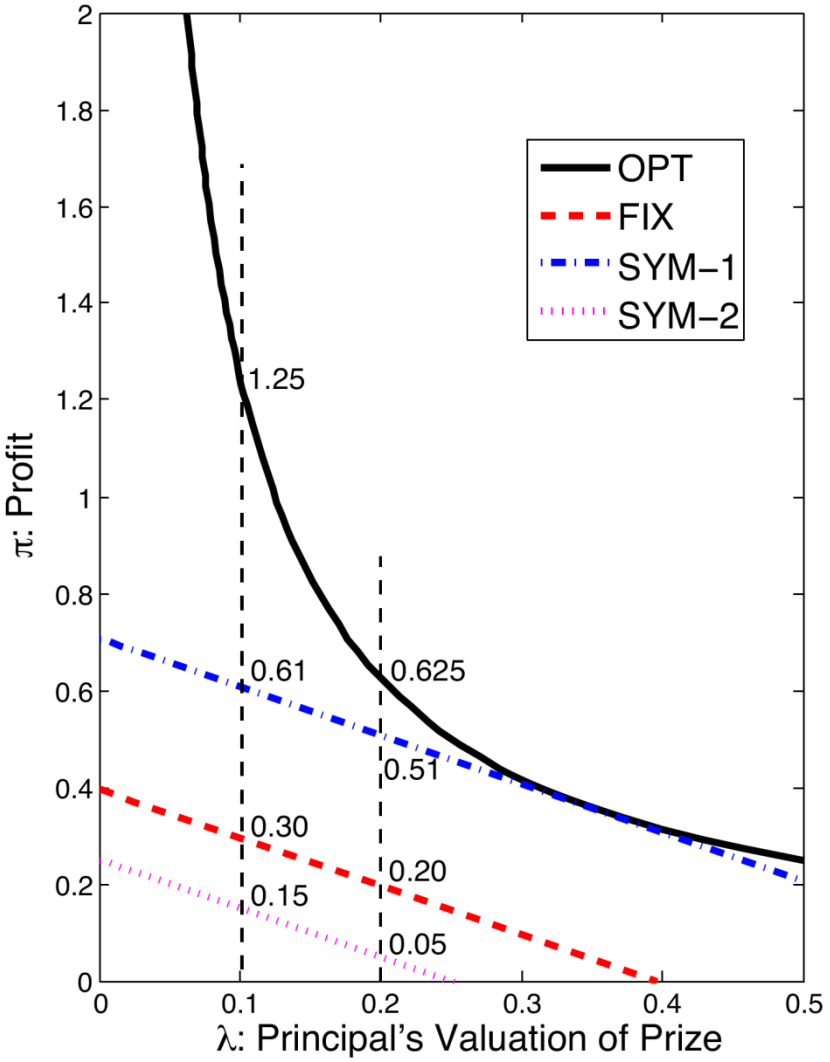


2. **Increases** crowdsourcing **revenue**
   - Overcomes <u>effort reservation</u>: fixed prize gives stronger agents incentive to reserve effort because a larger winning margin does not make a winner better off

3. **Enhances** system **scalability**
   - Overcomes <u>diminishing marginal return</u> (DMR) which is a universal law governing most economic phenomena



Num. Result 1: Profit Ranking



OPT: **all-pay auction with optimal prize tuple** (our mechanism)

FIX: fixed-prize, asymmetric

SYM-1: fixed-prize, symmetric – both agents follow $F_1(v)$ (stronger)

SYM-2: fixed-prize, symmetric – both agents follow $F_2(v)$ (weaker)

Result:

SYM-2 < FIX < SYM-1 < OPT

1) SYM-2 < FIX < SYM-1 : intuitive

- **SYM-2**: (weak, weak)
- **FIX**: (strong, weak)
- **SYM-1**: (strong, strong)

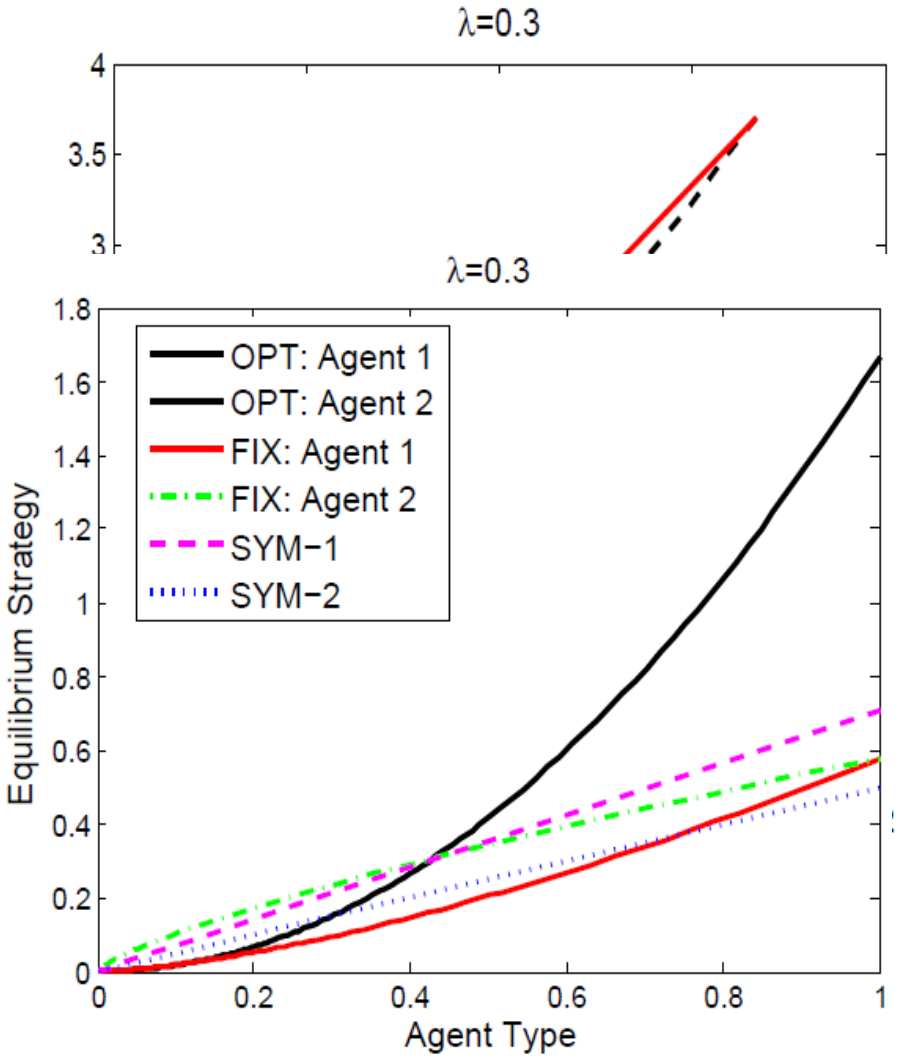
2) SYM-1 < OPT: puzzling

- **SYM-1**: (strong, strong)
- **OPT**: (strong, weak)

Answer lies in the optimal prize tuple

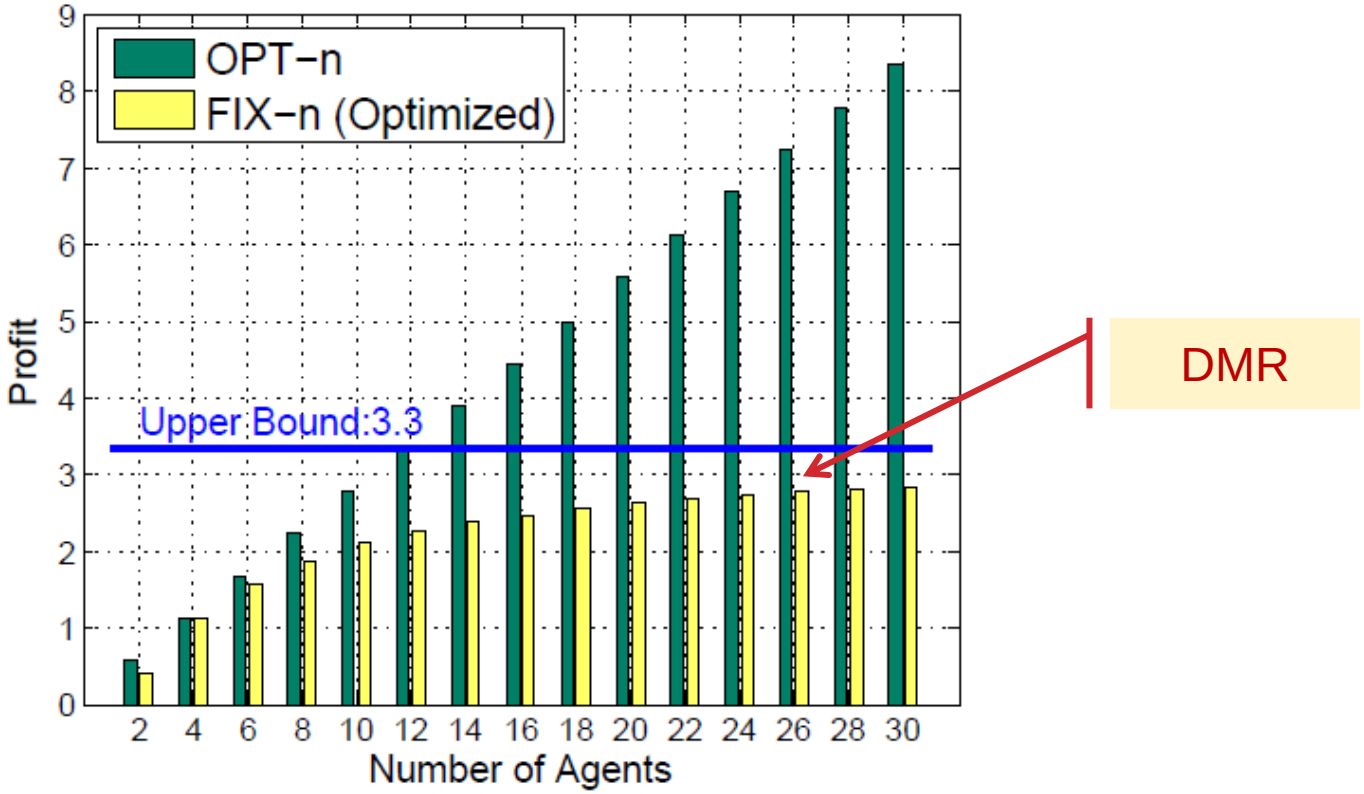
The prize tuple gives slightly **higher incentive to the weaker agent** (agent 2)

- This motivates agent 2 to work harder to compete with the stronger agent (agent 1)
- Agent 1 knows this (by reasoning) and hence will **not reserve effort** as in classic (fixed-prize) auctions



Numerical result 2: Scalability (SA negates DMR)

- **OPT-n**: OPT with *n* symmetric agents
- **FIX-n**: FIX with *n* symmetric agents



Recap

Heterogeneous players

- Modeled as asymmetric (all-pay) auction

Prize tuple for revenue maximization

Strategy autonomy (SA)

- symmetric equilibrium in asymmetric auction

Autonomy
iMANAGE

T. Luo, S. S. Kanhere, S. K. Das, and H-P. Tan, “Incentive mechanism design for heterogeneous crowdsourcing using all-pay contests”, IEEE Transactions on Mobile Computing (TMC), 2016.

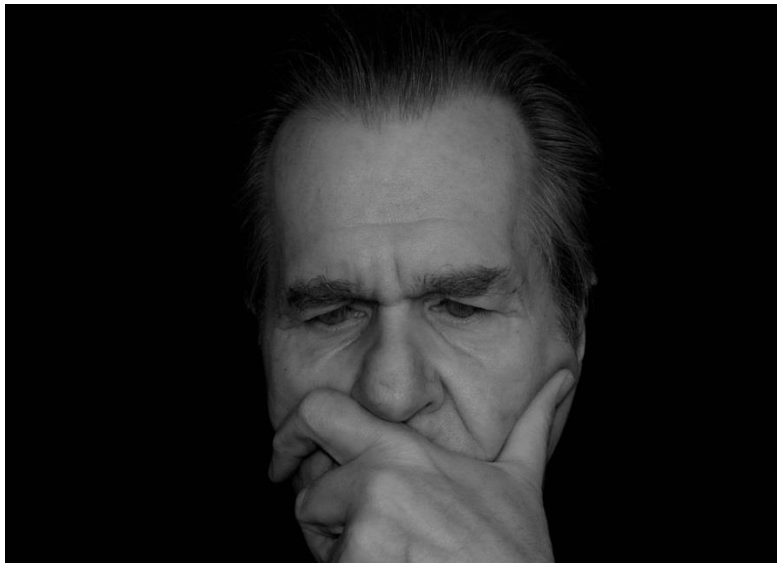
Outline

- Introduction

- **Incentives**
  - Fundamentals of mechanism design
    - Bayesian mechanism design
  - Crowdsourcing and All-pay auctions
  - **Tullock contests**

- Trust
- Privacy

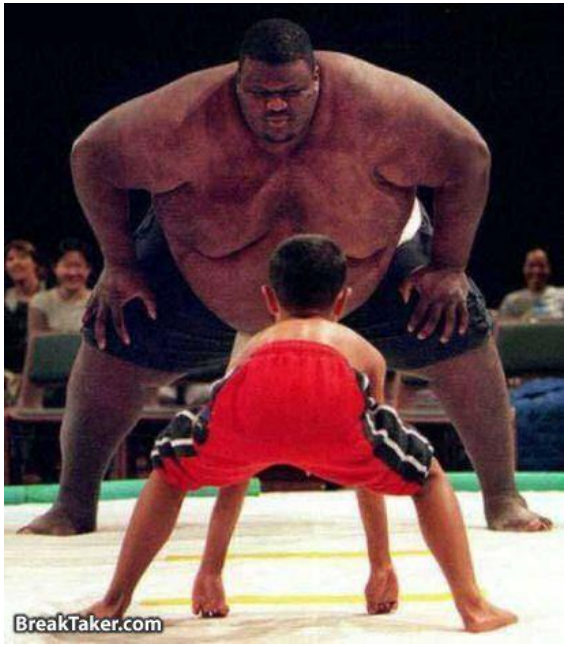
Taking a step back: Why auctions?



Pros and cons of auctions

Pros	Cons
• Well studied • Desirable properties (e.g. DSIC) • Classic mechanisms (e.g., VCG auctions) providing “templates”	• **Competitive**: must outbid everyone else in order to win • **Perfectly discriminating**: if you are not (among) the “strongest”, you lose for sure

Hard to attract a large number of participants



Alternative?

You always have a chance

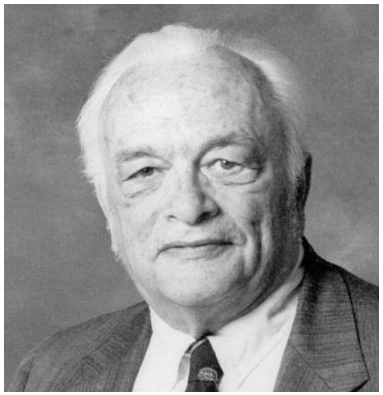
(no matter how weak you are)

Tullock contests

Imperfectly discriminating

- User-entry friendly

Gordon Tullock
(1922 – 2014)



Which is better?

Auctions	Tullock Contests
Suitable for **strong** players	Suitable for "**ordinary**" people
Fierce competition: tends to elicit the "**best**" contributions	**Lower barrier-to-entry**: conducive to population diversity and geographic coverage
Suitable applications: **effort- or knowledge-intensive** crowdsourcing	Suitable applications: **microtask** crowdsourcing
Revenue comparison: No conclusive result	

T. Luo, S. S. Kanhere, H-P. Tan, F. Wu, and H. Wu, "Crowdsourcing with Tullock Contests: A New Perspective", IEEE INFOCOM, 2015. (Best Paper candidate).

Fundamentals

Contest success function: $\Pr(b_i) = \frac{b_i^r}{\sum_{j=1}^{n} b_j^r}, \qquad r > 0$

- $r = 1$: Lottery

Payoff: $\Pr(b_i)V - c_i b_i$

Contest Model

Contest success function

- $\Pr(b_i) = \frac{g(b_i)}{\sum_{j=1}^{n} g(b_j)}$
- b_i : player effort
- $\xi_i = g(b_i)$: contribution ex: $g(b_i) = b_i$ (lottery)

Prize function

- $V(\xi_w)$ where ξ_w is winner's contribution

Player payoff

- $u_i = \Pr(b_i)\, V(\xi_w) - c_i b_i$ ← maximize
- c_i : marginal cost (type); every player only knows his own type

Organizer profit

- $\pi = \nu \sum_{i=1}^{n} \xi_i - V(\xi_w)$ ← maximize
- ν : organizer's valuation of per unit user contribution

- Even the simplest, conventional Tullock contest is analytically intractable (because of double uncertainty)

- We managed to obtain a simple, and in most cases closed-form, solution

Main Result

Optimal prize function:

$$V^*(\xi_w) = \left[\beta^{-1}(\xi_w)h(\xi_w) - \int_{\underline{\xi}}^{\xi_w} h(\tilde{\xi})\,\mathrm{d}\beta^{-1}(\tilde{\xi})\right] \Big/ p(\xi_w)$$

Player strategy:

$$h'(\xi) = \frac{\nu}{c + \frac{F(c)}{f(c)}}$$

Maximum profit:

$$\pi^* = n \int_{\underline{c}}^{\overline{c}} \left[\nu\beta(c) - h(\beta(c))c + \frac{F(c)}{f(c)}[h(\beta(\overline{c})) - h(\beta(c))]\right] \mathrm{d}F(c)$$

Qualitative properties

Closed-form player strategy in most cases

- Well suited for rapid prototyping on smartphones & wearable widgets

Everyone contributes

- In contrast to auctions

Strategy oblivious to n

- Overcomes disincentive when there are more competitors


Performance

- Optimize conventional Tullock contests
- Prior work only analyzes conventional Tullock contests

Problem formulation:

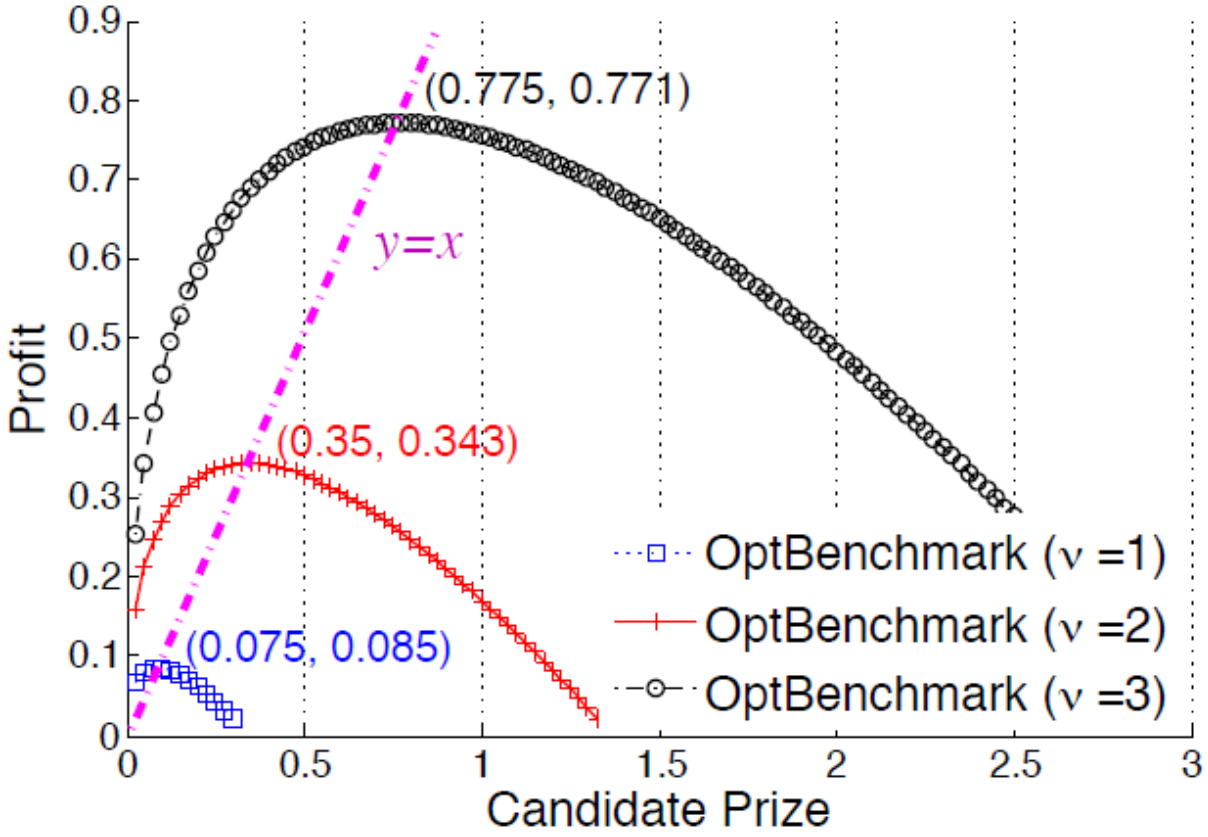
- Maximize profit: $n\nu \int_{\underline{c}}^{\overline{c}} \beta_0(c)\,\mathrm{d}F(c) - V_0.$
- where strategy β_0 is determined by

$$\int_{\Theta^{n-1}} \frac{\sum_{j=1}^{n-1} \beta_0(\tilde{c}_j)}{[\beta_0(c) + \sum_{j=1}^{n-1} \beta_0(\tilde{c}_j)]^2} \prod_{j=1}^{n-1} \mathrm{d}F(\tilde{c}_j) = h'(\xi_0)\frac{c}{V_0}$$

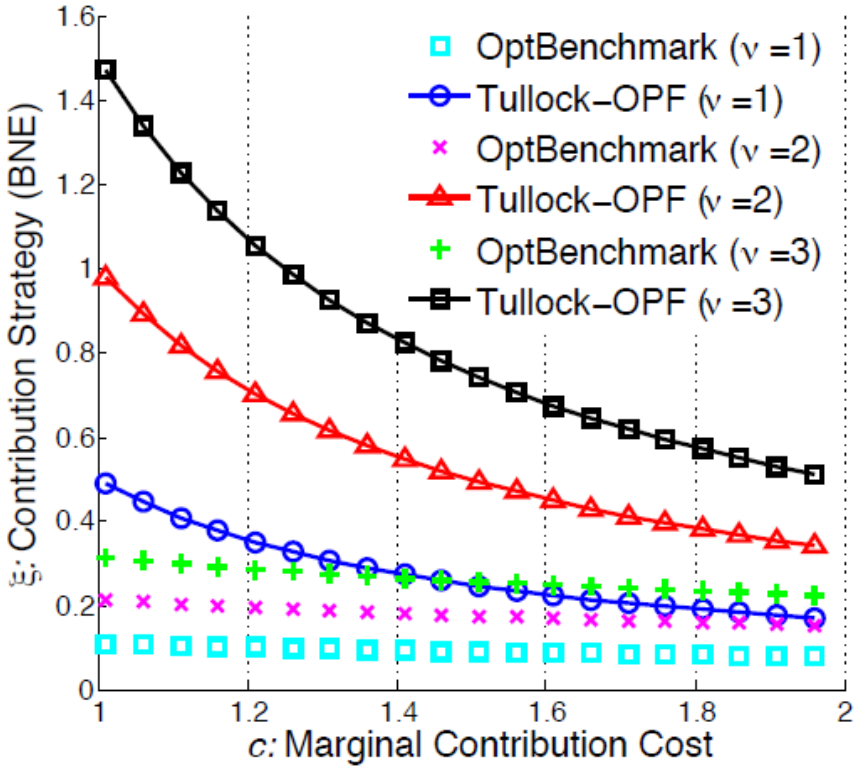
Construct optimal benchmark

Numeric method

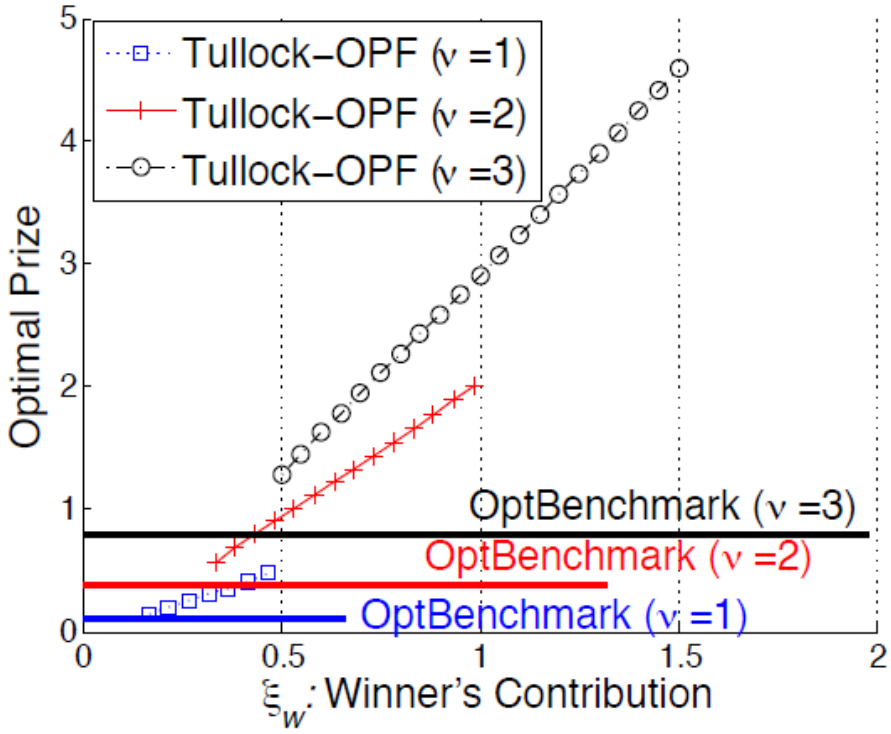
- Fredholm *equations*



Revenue



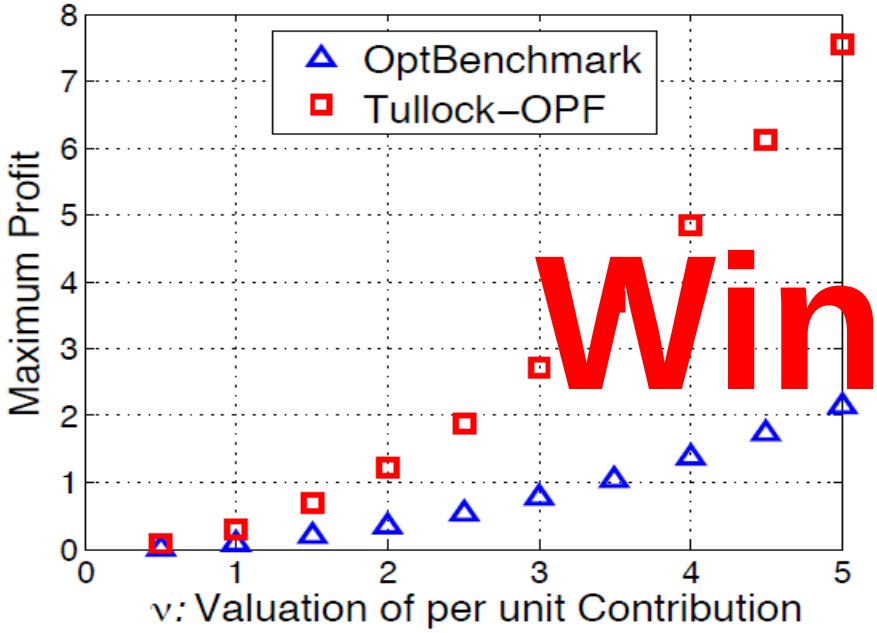
Cost



Higher revenue yet higher cost

Everyone contributes

Profit



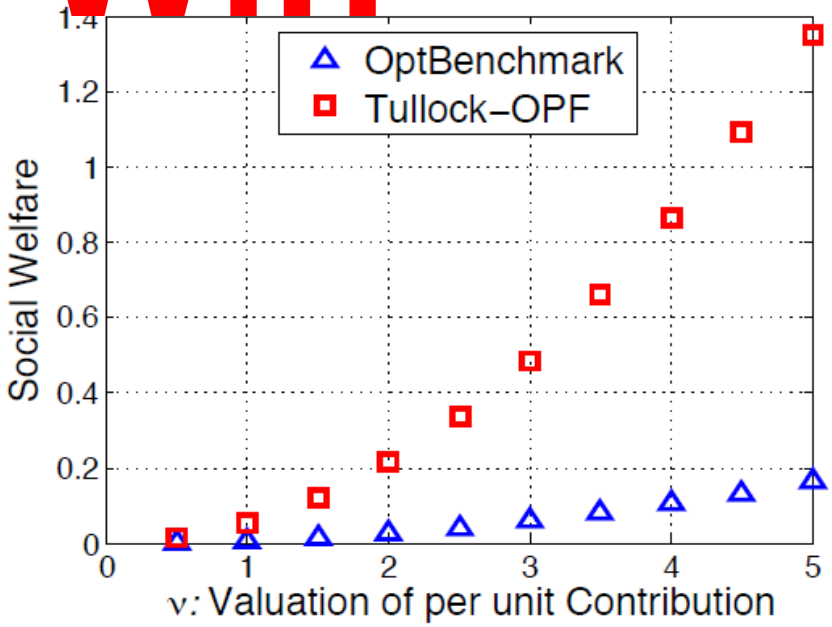
Organizer's **Profit**:

~3.5 times of benchmark

Social welfare

Players' **Payoff**:

7-9 times of benchmark



Win-Win

Tullock contests

You always have a chance

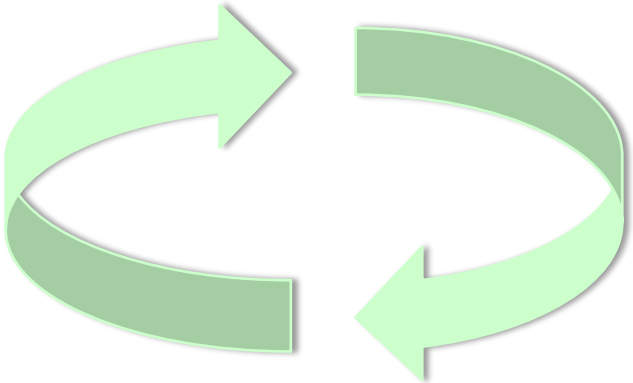
Summary

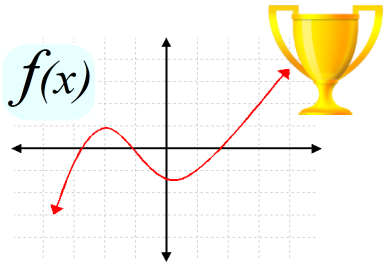
1

Bayesian mechanism design
Incomplete information

2

Winner-pay auctions





4

Tullock contests

Heterogeneous all-pay

3

PART II.
TRUST

Outline

- Introduction
- Incentives
- **Trust**
  - **Motivating experiment**
  - Reputation framework
  - A social-network perspective
- Privacy
- Conclusion

Motivating Experiment

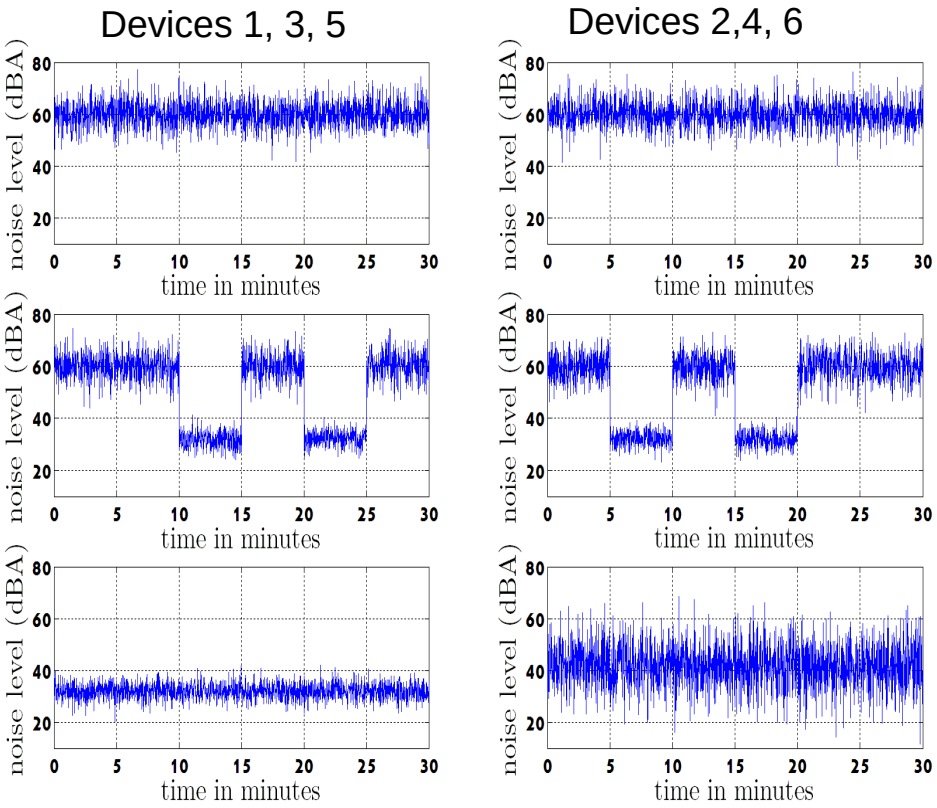
Objective

- using mobile phones to determine the average noise level in our office

Setup

- 6 iPhones are instrumented as SLM, collecting samples over 30 minutes.
- devices are selectively placed in the drawers (simulate bad contributions)
- devices 1 and 2 were kept on the desks
- devices 3 and 4 were toggled between desks and drawers
- device 5 was placed in the drawer at all time
- device 6 was placed in the drawer at all time with random noise added

OBSERVATIONS



Simple averaging?

- include erroneous data

Weighted averaging?

- each dev. is assoc. with a weight
- weight reflects the data quality

Problem

- ground truth information not available
- How to determine weights?

Outlier Detection

Principle

- group consensus is obtained from all devices
- distances to the consensus determine the weights
- distances & weights are inversely proportional

Consider

- device 5 at the 5^{th} time epoch
- server deduces device 5 is bad by comparing with contributions from devices 1 to 4
- server thus assigns lower weight to device 5's data

Outlier Detection (contd.)

- outlier detection treats each epoch independently of each other
- it is not possible to gain insight into the long-term device behaviour
- long-term information is valuable in reinforcing confidence
- Analogy: human behaviour

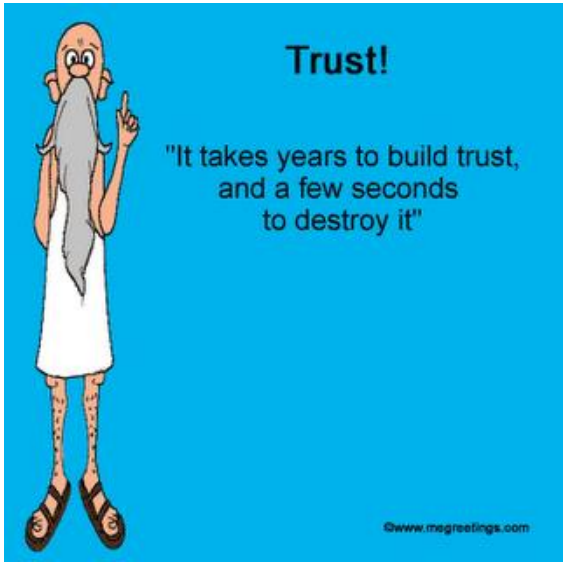
What do humans do?

We use the concept of reputation

It’s an asymmetrical construct

→ slowly accumulate trust with positive experiences

→ rapidly tear down trust with a small # of negative experiences



Need for a Reputation Framework

Consider

- device 5 at the 5th epoch as before
- server keeps track of the behaviour of this device since the 1st epoch
- continuous bad data -> progressively lower weight
- progressively lower weight -> more accurate approximations

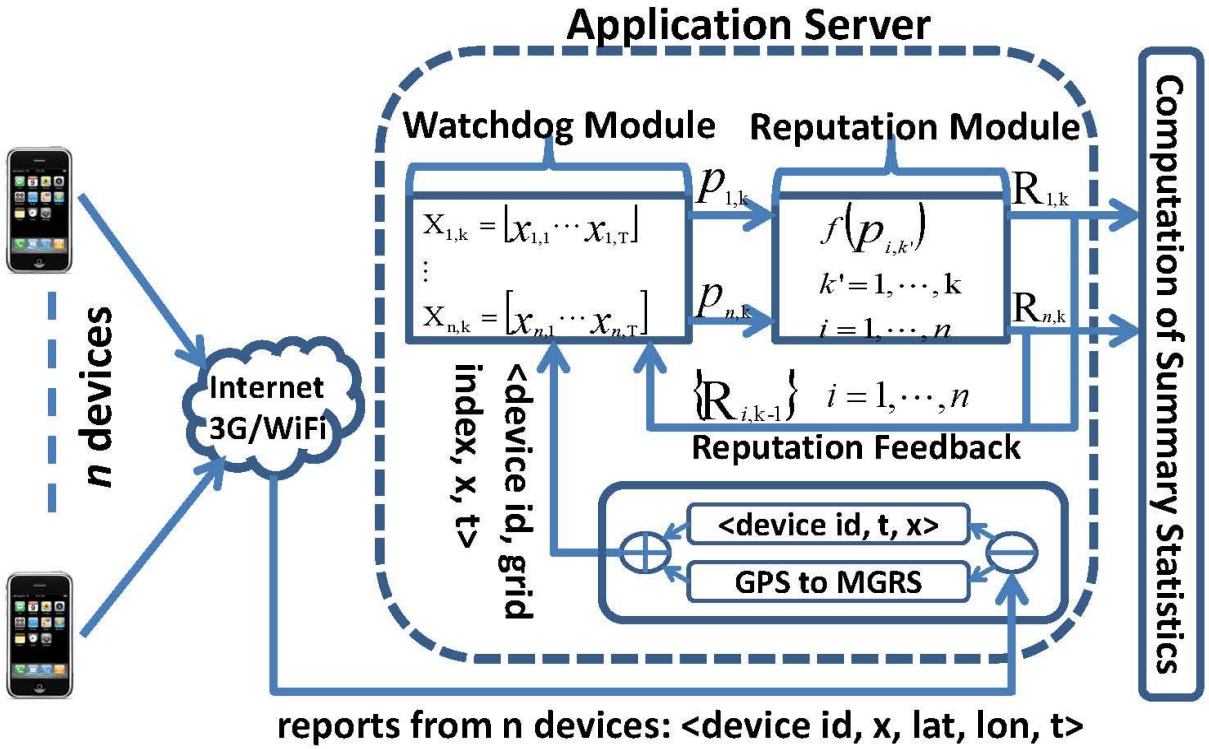
Looking into the past and incorporating historical info

Reputation System

Roadmap

- Introduction
- Incentives
- **Trust**
  - Motivating experiment
  - **Reputation framework**
  - A social-network perspective
- Privacy
- Conclusion

System Architecture



K. Huang, S. S. Kanhere and W. Hu, On the Need For a Reputation System in Mobile Phone Based Sensing, in Ad Hoc Networks, vol. 12, pp. 130-149, January 2014.

System Description

Two main components

- watchdog module: outlier detection -> instantaneous view of devices
- reputation module: reputation function -> long-term view of devices

Spatial & Temporal segmentation

- data are considered together only if they are contextually related
- E.g., noise samples measured at the same time, but 100 meters apart?
- E.g., noise samples measured at the same place, but an hour apart
- Spatial dimension -> grids; Temporal dimension -> time epochs
- Granularity of time and space is application-specific

Building Blocks

Watchdog Module

- provides instantaneous view about devices
- processes user data in epochs of duration T
- implements a consensus-based, iterative outlier detection algorithm
- outputs a set of device cooperative ratings {0,1}
- rating of value > 1/n (n = total # of devices) -> cooperative behaviour

WATCHDOG MODULE (CONTD.)

* **instantaneous average**

$$r_t = \sum_{i=1}^{n} p_{i,k} x_{i,t},\ (k-1) \times T < t \leq k \times T$$

* **device rating**

$$p_{i,k} = \frac{\frac{1}{\frac{\sum_{t=1}^{T}(x_{i,t}-r_t)^2}{\sum_{i=1}^{n}\sum_{t=1}^{T}(x_{i,t}-r_t)^2}+\epsilon}}{\sum_{j=1}^{n}\frac{1}{\frac{\sum_{t=1}^{T}(x_{j,t}-r_t)^2}{\sum_{i=1}^{n}\sum t{=}1^{T}(x_{i,t}-r_t)^2}}} + \epsilon$$

*** Convergence condition**

$$|p_{i,k}^{l} - p_{i,k}^{l-1}| < 0.0001$$

Robust average computation

Let $p_{i,k}^{l}$ and r_t^{l} be the values of $p_{i,k}$ and r_t at the l^{th} iteration, respectively

1. Initialize $l = 0$ and $p_{i,k}^{l} = \frac{1}{n}$
2. Compute r_t^{l+1} from $p_{i,k}^{l}$ using Eq. 2
3. Compute $p_{i,k}^{l}$ from r_t^{l} using Eq. 3
4. $l \leftarrow l+1$
5. Start from Step 2 if no convergence

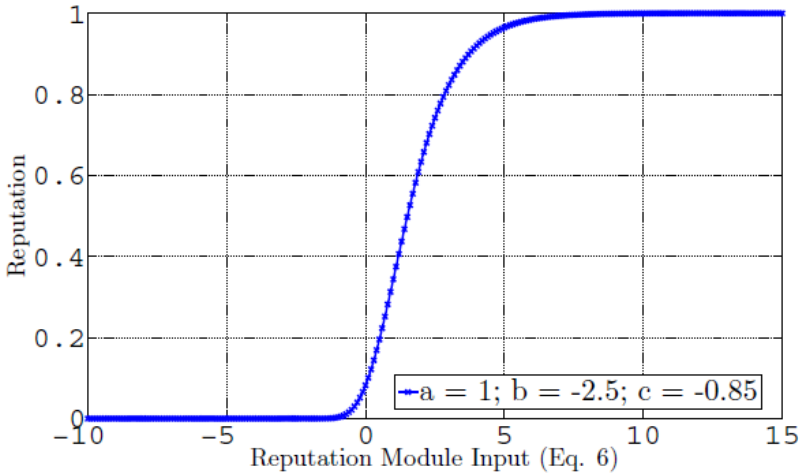
C. Chou, A. Ignjatovic and W. Hu, "Efficient Computation of Robust Average in Wireless Sensor Networks using Compressive Sensing", Tech. Report #915, UNSW, 2009.

Building Blocks (contd.)

Reputation module

- builds long-term perspective about device trustworthiness
- takes as inputs, past device cooperative ratings
- applies Gompertz function to produce reputation scores
- outputs a set of reputation scores {0,1}

REPUTATION MODULE (IN DEPTH)



Gompertz function is used

$$R_{i,k}(p'_{i,k}) = ae^{be^{cp'_{i,k}}}$$

input: cooperative ratings

output: reputation scores

fast deterioration of reputation

slow build-up of reputation

Rep. module needs to address

- accumulation of historical info
- most recent info -> more relevant
- Input spans to –ve value

Our solution

- normalizing coop. ratings to {-1,1}
- set input of Gompertz func as

$$p'_{i,k} = \sum_{k'=1}^{k} \lambda^{(k-k')} p^{norm}_{i,k'}$$

- using "lamda" as ageing weights
- two different values for "lamda" based on coop. or non-coop. behaviours

Uses of Reputation Scores

As a weight associated with user contributions

- example: more accurate summary statistics such as average.

$$\bar{x_t} = \sum_{i=1}^{n} R_{i,k} \times x_{i,t},\ (k-1) \times T < t \leq k \times T$$

As a filter to select user contributions

- server only accepts data from devices with good prior reputation
- server revokes all devices that cannot be trusted
- revocation establishes a feedback link from REP to WD modules
- revocation is server-wide -> excludes devices from both WD and REP
- revocation prevents the propagation of errors
- our implementation prevents devices from being infinitely revoked

INFINITE REVOCATION

Feedback

- app. uses rep. from t-1 to revoke devices in t

What happens in t and t+1?

- revocation is server-wide
- devices removed from WD
- device reps. undergo ageing @ t
- R(t) < R(t-1)
- at t+1, devices are revoked again

Two-pass outlier detection

- separate devices into {reputable} & {disreputable} @ time t
- run the robust algorithm for the 1st time with {reputable} set to obtain the robust average
- compare data from {disreputable} set with the robust average
- move {disreputable} to {reputable} if the difference is within range
- run the robust algorithm for the 2nd time with expanded {reputable} set

Experiment Evaluation

Objective

- computing the average noise level in the main library of UNSW (EarPhone)
- exercising rep. system with inadvertent and malicious users
- comparing the performance with state-of-the-art Beta rep. system

Equipment

- 8 Apple iPhones running off-the-shelf ‘SPL Graph’ application
- Centre 322 SLM and Data Logger (used to collect ground truth)

Spatial & Temporal segmentations

- Grid size = 30*m* by 30*m,* in accordance with Australian Acoustic Standards
- Temporal epoch = 1 minute

Procedures

Cooperative vs. non-cooperative behaviours

- cooperative -> users expose the microphone of the phone
- non-cooperative -> users place the phones in pant pockets

Consider 3 scenarios

- 1st: w/o malicious users; phones are changed randomly every 10 mins.
- 2nd: w/o malicious users; phones are changed randomly every min.
- 3rd: with malicious users; phones are changed randomly every min.

Comparisons & Metrics

We compare different types of averages

- Raw -> simple averaging without any associated weights
- Robust -> weighted averaging (weights = output of WD = coop. ratings)
- Beta -> weighted averaging (weights = output of rep. = reputation scores)
- Gompertz with and w/o feedback link -> weighted averaging

Evaluation metrics

- we use mean root mean square error (RMSE) w.r.t the ground truth
- percentage of epochs in which Gompertz rep. outperforms Beta rep.

SCENARIO 1: EVOLUTION OF REPUTATION SCORES

1: non-cooperative, 0:cooperative

device 1	1	0	0	0	1	1
device 2	1	1	1	0	0	0
device 3	0	0	0	1	1	1
device 4	0	1	0	1	0	0
device 5	1	1	1	1	1	1
device 6	0	0	0	0	0	0
device 7	0	0	1	0	0	0
device 8	0	0	0	0	0	0

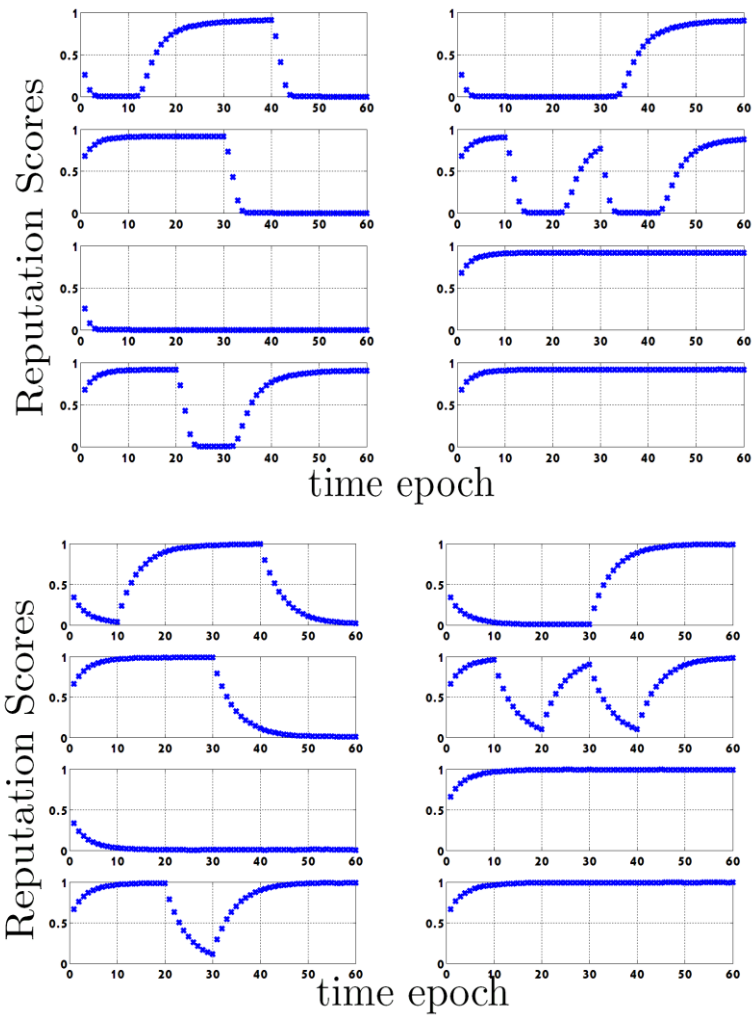
app. does not know the true pos. of devices

Gompertz rep. scores (top) successfully track device behaviour and quicker than Beta reputation (bottom)

Reputation increases gradually but decreases rapidly

Devices 1, 3, 5

Devices 2,4, 6



SCENARIO 1: COMPARISONS

Time to learn

- time delay incurred in adjusting to changes in user behaviours, e.g., t = 20
- Gompertz rep. takes relatively short time to learn about these events (e.g., 3 min)

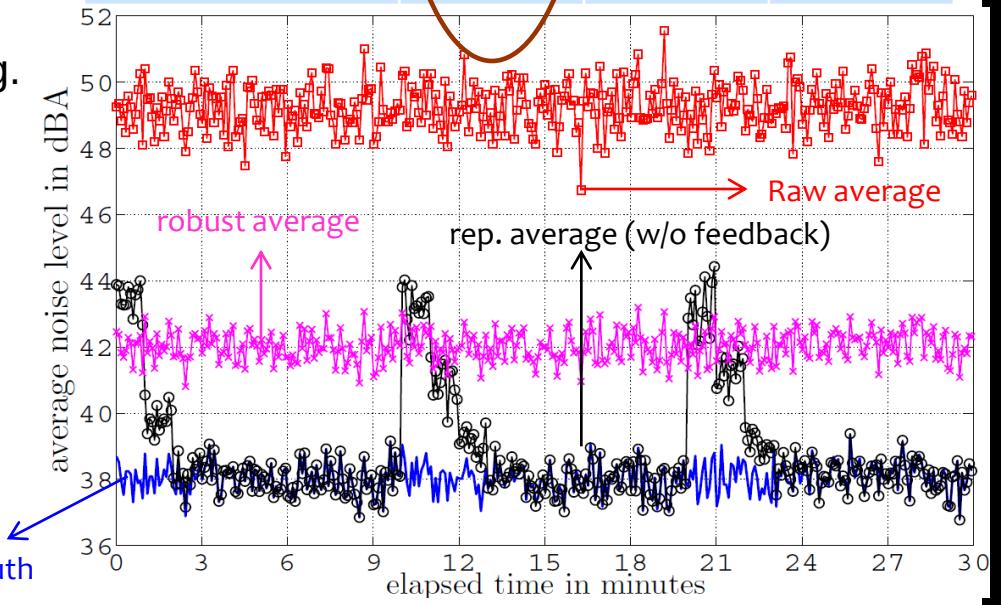
Better performance

- Gompertz results in the best estimation
- improved by a factor of 5 w.r.t robust avg.
- improved by a factor of 3 w.r.t Beta avg.

Benefit of Reputation Feedback

- an extra 53% reduction compared to the non-feedback configuration.

Type of Average	scenario 1	scenario 2	scenario 3
Raw	11.28	8.39	8.23
Robust	4.02	3.90	4.29
Beta	2.33	3.88	4.27
Gompertz (w/o f-b)	0.73	2.68	3.73
Gompertz (w f-b)	0.34	1.76	2.08



Type of Comparison	scenario 1	scenario 2	scenario 3
Gompertz > Beta	100%	87%	86%
Gompertz (w f-b) > Beta	100%	88%	92%

Percentage of epochs where Gompertz > Beta

SCENARIO 2

Type of Average	scenario 1	scenario 2	scenario 3
Raw	11.28	8.39	8.23
Robust	4.02	3.90	4.29
Beta	2.33	3.88	4.27
Gompertz (w/o f-b)	0.73	2.68	3.73
Gompertz (w f-b)	0.34	1.76	2.08

more frequent changes in device positions (every minute)

statistical worst-case scenario for the reputation system

Gompertz outperforms Beta by 30% (w/o feedback) and 54% (with feedback)

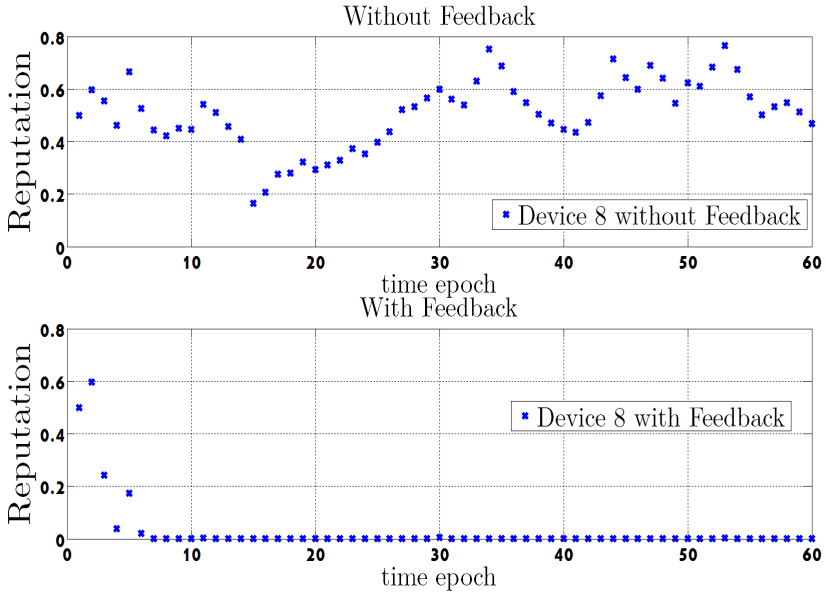
Gompertz outperforms Beta in 52 out of 60 epochs (around 88%)

Type of Comparison	scenario 1	scenario 2	scenario 3
Gompertz > Beta	100%	87%	86%
Gompertz (w f-b) > Beta	100%	88%	92%

Percentage of epochs where Gompertz > Beta

Scenario 3

Type of Average	scenario 1	scenario 2	scenario 3
Raw	11.28	8.39	8.23
Robust	4.02	3.90	4.29
Beta	2.33	3.88	4.27
Gompertz (w/o f-b)	0.73	2.68	3.73
Gompertz (w f-b)	0.34	1.76	2.08



malicious behaviour is considered

1st type: constant offset of 30dB (dev. 7)

2nd type: random Gaussian offset (dev. 8)

Gompertz leads Beta in 86% (w/o feedback) and 92% (with feedback) of epochs

feedback config. incurs smaller penalty (18%) than non-feedback config (39%)

feedback config. is robust to both types of malicious behaviour

Type of Comparison	scenario 1	scenario 2	scenario 3
Gompertz > Beta	100%	87%	86%
Gompertz (w f-b) > Beta	100%	88%	92%

Percentage of epochs where Gompertz > Beta

Key Contributions & Results

We made the case for using reputation system to evaluate device trustworthiness in the context of participatory sensing

Proposed the use of Gompertz function to compute device reputation scores

Evaluated system performance in the context of real-world participatory sensing application

Demonstrated the superior performance of Gompertz reputation system over the current state-of-the-art

Demonstrated the benefit of revoking disreputable devices, i.e., establishing a feedback in the reputation system

Reputation framework is generic and can work with a variety of participatory sensing applications

Outline

- Introduction
- Incentives
- **Trust**
  - Motivating experiment
  - Reputation framework
  - **A social-network perspective**
- Privacy
- Conclusion

Trust:
a Social-network perspective

Basic idea

Common assumption in prior art: people are **self-interested**

Humans are **multi-facet**; sometimes they are **altruistic**

A plausible motivation could be

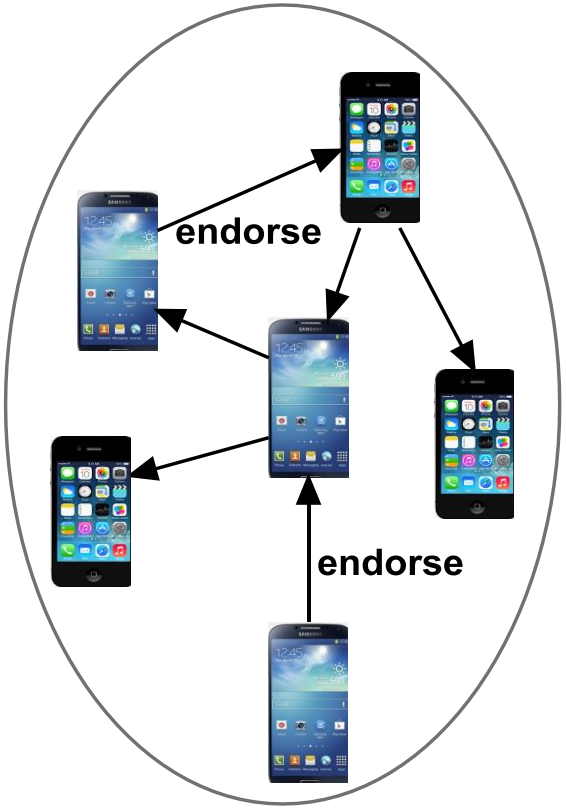
"Work for your cared / loved ones" (besides yourself)

Simple Endorsement Web (SEW)

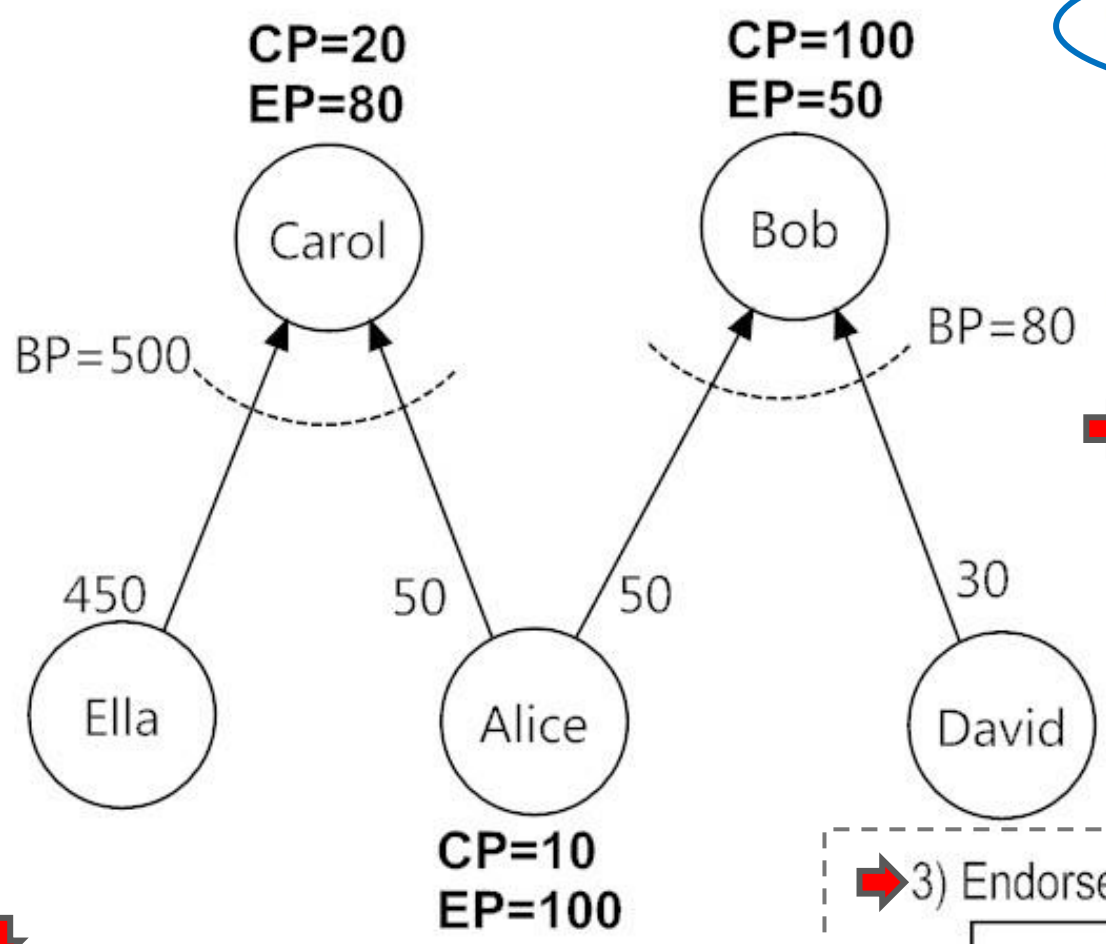
A can **endorse** B if

- A *trusts* B, or
- B *cares about* A (nepotism)

A is a beneficiary of B



T. Luo, S. S. Kanhere, and H-P. Tan, “SEW-ing a Simple Endorsement Web to incentivize trustworthy participatory sensing”, IEEE SECON, 2014.



$$EffectP = CP(1+\frac{BP}{BP+CP})$$

$$EffectP_{Bob} = 100(1+\frac{80}{80+100}) = 144$$

$$EffectP_{Carol} = 20(1+\frac{500}{500+50}) = 38$$

1) Contributors' reward (CP grows):

$$CP_{Bob} \uparrow: \frac{144}{144+38}R = 0.8R$$

$$CP_{Carol} \uparrow: \frac{38}{144+38}R = 0.2R$$

2) Appreciation power for contributors:

$$AP_{Bob} = \frac{144-100}{144} \times 0.8R = 0.24R$$

$$AP_{Carol} = \frac{38-20}{38} \times 0.2R = 0.09R$$

3) Endorsers' reward (EP grows):

$$EP_{David} \uparrow: \frac{30}{80} AP_{Bob} = 0.09R$$

$$EP_{Ella} \uparrow: \frac{450}{500} AP_{Carol} = 0.08R$$

$$EP_{Alice} \uparrow: \frac{50}{80} AP_{Bob} + \frac{50}{500} AP_{Carol} = 0.16R$$

Power redemption: Stackelberg game

Redemption: $$$ = CP x α + EP x β, $\alpha > \beta > 0$

Stackelberg game:

- **Leader**: organizer announces exchange rates α, β
- **Follower**: each participant determines contribution quality

Participant's utility (two components):

$$\text{Contributing}: \quad u_i^c = x_c r_i - c_i(q_i),$$

$$\text{Endorsing}: \quad u_i^e = x_e \sum_{k \in \mathcal{N}_i^{out}} \eta_{ik} \rho_k r_k.$$

Organizer's utility:

$$\text{maximize: } x_0 \log(1 + \sum_{k \in \mathcal{N}} q_k^*) - x_c R\left(1 + \epsilon \sum_{k \in \mathcal{N}} \rho_k \frac{q_k^* P_k}{\sum_{j \in \mathcal{N}} q_j^* P_j}\right)$$

Optimal exchange rate in equilibrium ($\beta = \epsilon\, \alpha$):

$$\alpha = \frac{x_0}{(1+\epsilon\rho)R} - \frac{nt}{(n-1-\epsilon\rho)R}$$

Optimal endorsing strategy

Whom to endorse?

Algorithm 1 Endorser's decision: Constructing the optimal portfolio of contributors

Input: $\mathcal{C}, \mathcal{N}^{soc}$
Output: $\mathcal{N}^*_{out}$

$\mathcal{N}^* \leftarrow \emptyset$, $Y_{max} \leftarrow \sum_{k\in\mathcal{N}^{soc}} AP^+_k|_{\tilde{n}=0}$
for $\tilde{n} = 1 \rightarrow n_1$ **do**
  Compute $AP^+_k(\tilde{n})$ for all $k \in \mathcal{C}$ using (20)
  $\mathcal{L}^{soc} \leftarrow \{AP^+_k(\tilde{n})|k \in \mathcal{N}^{soc}\}$,
  $\overline{\mathcal{L}^{soc}} \leftarrow \{AP^+_k(\tilde{n})|k \in \overline{\mathcal{N}^{soc}}\}$
  $\mathcal{L}^{ind} =$`PartialQuickSort`$(\overline{\mathcal{L}^{soc}}, 1, n_1, \tilde{n})$
  $Y_{\tilde{n}} \leftarrow \sum_{i=1}^{n_0} \mathcal{L}^{soc}[i] + \sum_{i=1}^{\tilde{n}} \overline{\mathcal{L}^{soc}}[i]$
  if $Y_{\tilde{n}} > Y_{max}$ **then**
    $Y_{max} \leftarrow Y_{\tilde{n}}$
    $\mathcal{N}^* \leftarrow \mathcal{L}^{ind}[1..\tilde{n}]$
  end if
end for
return $\mathcal{N}^* \cup \mathcal{N}^{soc}$

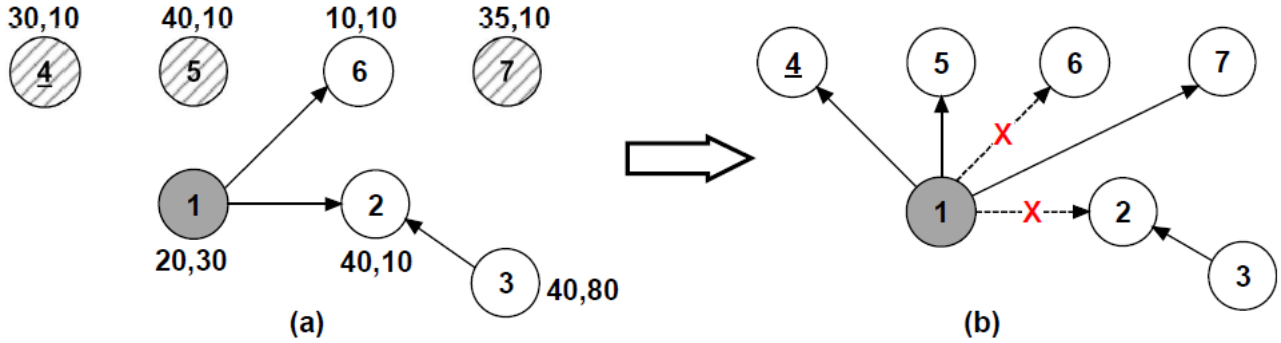
Whom to endorse me?

Algorithm 4 Contributor's decision: Constructing the optimal portfolio of endorsers

Input: $\mathcal{D}, \mathcal{N}^{bnf}, \mathring{\mathcal{N}}^{bnf}, \eta^{min}$
Output: $\mathcal{N}^*_{in}$

Compute $\delta^+_{EP,i}$ for all $i \in \mathcal{D}$
$\delta^{min}_{EP} \leftarrow \min_{j\in\mathring{\mathcal{N}}^{bnf}} \delta^+_{EP,j}$
$BP^{all} \leftarrow \delta^{min}_{EP}/\eta^{min}$
$BP^{cap} \leftarrow BP^{all} - \sum_{i\in\mathcal{N}^{bnf}} \delta^+_{EP,i}$
if $BP^{cap} \geq 0$ **then**
  return $\mathcal{N}^{bnf} \cup$ `Knapsack`$(BP^{cap}, \delta^+_{EP,i}|_{i\in\mathcal{D}\setminus\mathcal{N}^{bnf}})$
else
  return Nil
end if

An illustration



Candidate Sets	$\tilde{\pi}_k$	$\sum_k \tilde{\pi}_k$
$\{k\|\underline{4}, 5, 7\}$	$\{\underline{7.5}, 8, 7.78\}$	23.3
$\{k\|\underline{4}, 5, 6, 7\}$	$\{\underline{6}, 6.31, 4.28, 6.18\}$	22.8
$\{k\|2, \underline{4}, 5, 6, 7\}$	$\{1.9, \underline{5}, 5.22, 3.75, 5.12\}$	21.0

Node 1 receives requests from 4,5,7 to endorse them

Node 2 has higher CP than node 7, but loses in the competition

T. Luo, S. S. Kanhere, and H-P. Tan, “SEW-ing a Simple Endorsement Web to incentivize trustworthy participatory sensing”, IEEE SECON, 2014.

PART III.
PRIVACY

Roadmap

- Introduction
- Incentives
- Trust
- **Privacy**
  - **Collaborative path hiding**
  - AnonySense: Anonymous Tasking and Reporting
  - Private Data Vaults: Access Control
  - IncogniSense: Balancing Privacy and Trust
- Conclusion

Privacy Challenges

How do we obtain sensor data from users while protecting their privacy?

Just hiding identity is not sufficient since multiple reports may be linked as being from the same user and thus reveal user's identity

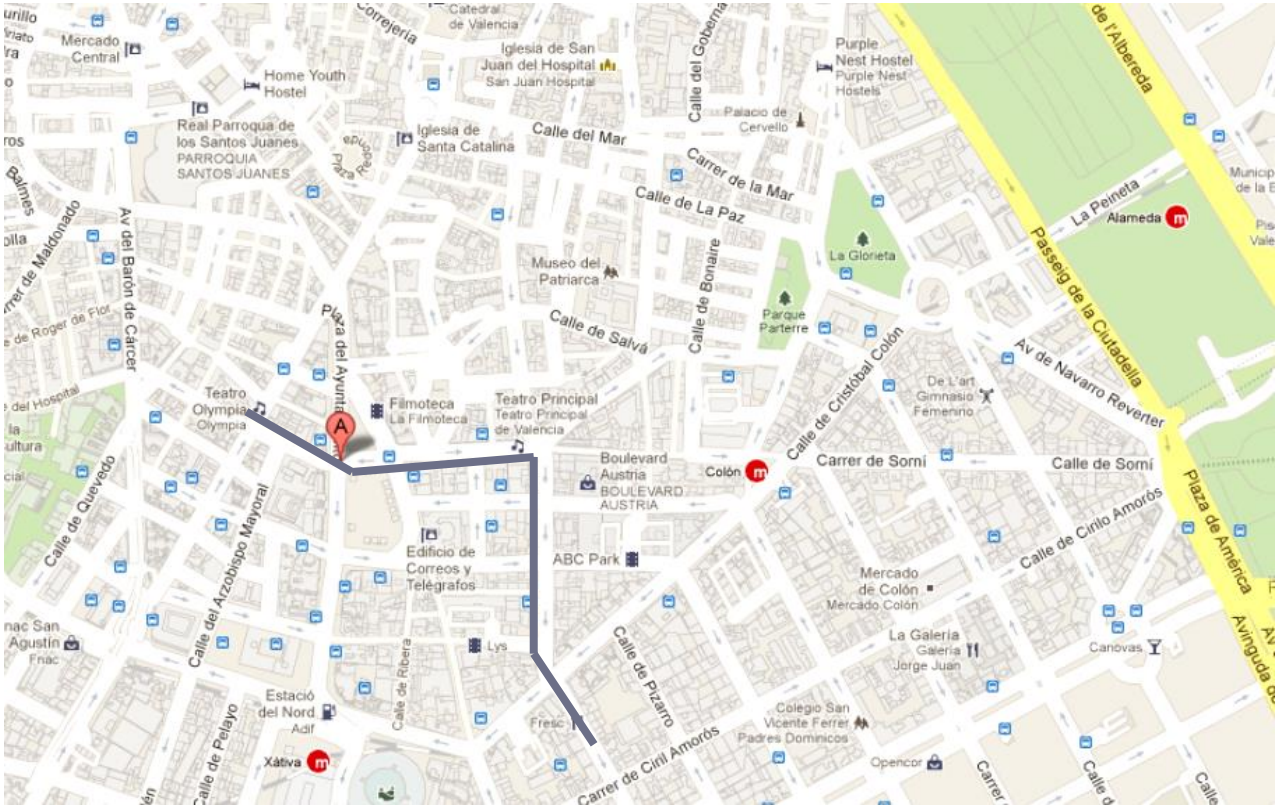
How do we empower users to control access to their personal data?

Privacy Challenges

Each sensor reading is uploaded with spatiotemporal metadata to the central server

A

A_1
A_2
A_3
A_4
A_5
A_6



Privacy Challenges

Adversary can infer frequently visited locations (home/work)

Simple techniques such as using pseudonyms or suppressing user identity may not always work

Adversary may have background information to link a user to their reports

Protecting privacy is critical as users would otherwise not be motivated to contribute data

State of the Art

Spatial Cloaking

The real location of the participants is replaced by the averaged location of k nearest participants

Cloaked location is determined by a central third party, which requires the location of all participants

Participants must trust this entity:

- not to breach their privacy
- apply efficient mechanisms to protect their privacy

Collaborative Path Hiding

No dependence on a central entity to protect the privacy of the participants

Give the control over their privacy to the participants

Decentralized and collaborative mechanism

D. Christin, J. Guillemet, A. Reinhardt, M. Hollick, S. S. Kanhere, “Privacy Preserving Collaborative Path Hiding for Participatory Sensing Applications”, in Proceedings of IEEE MASS 2011, Valencia, Spain, October 2011.

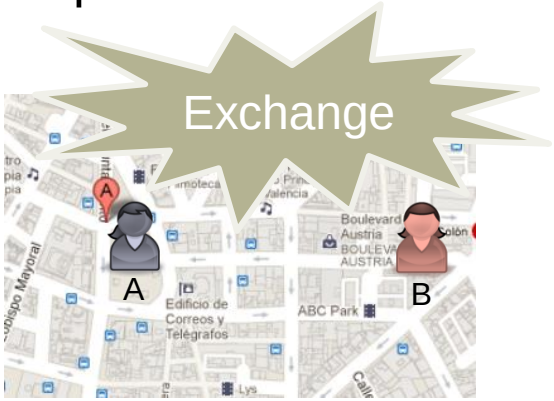
Path Jumbling Concept

Objective

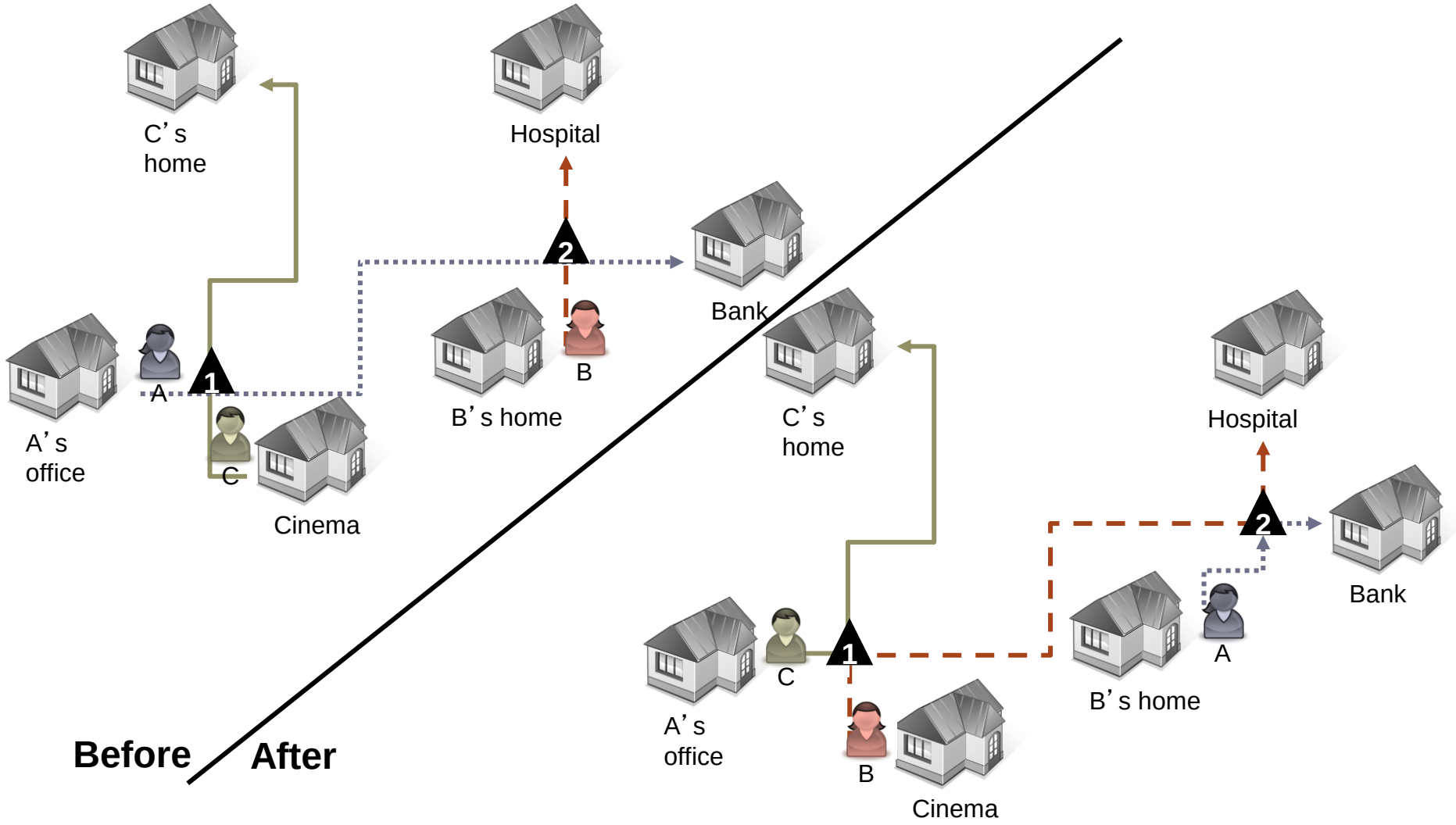
Break the link between the spatiotemporal context of the sensor readings and the identity of the participants

Method

Participants in physical proximity exchange the sensor readings including spatiotemporal metadata



Resulting Paths



Selected Exchange Strategies

Exchange strategies

How many sensor readings to exchange?

Which sensor readings to exchange?

Tradeoff between privacy protection against malicious participants and against malicious applications

Reporting strategies

When to report the sensor readings to the application server?

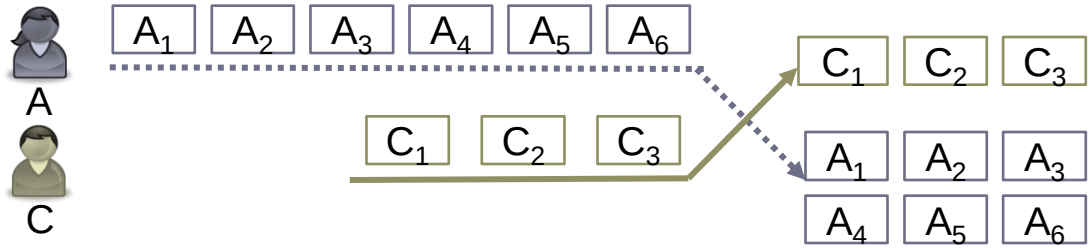
Tradeoff between timely delivery of the sensor readings to the application and privacy protection

Design Space

Realistic

Complete

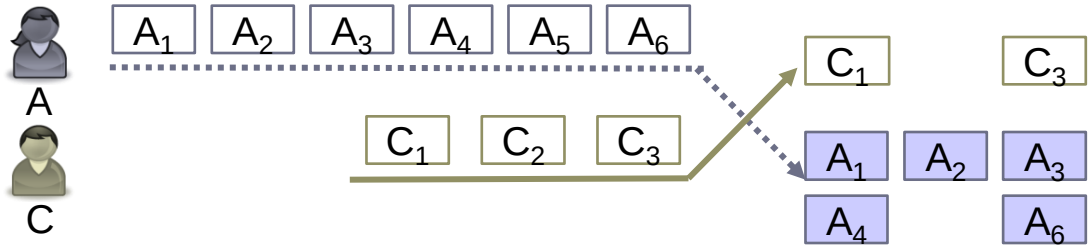
Asymmetric



Random-unfair

Partial

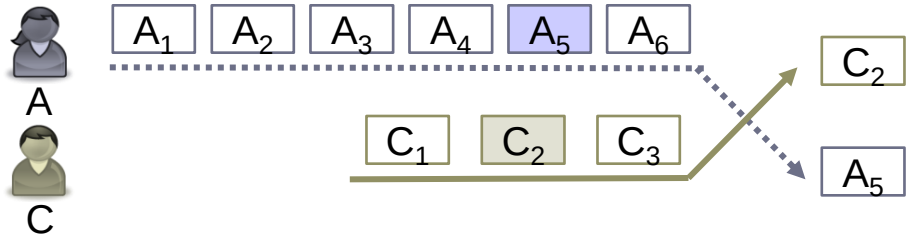
Asymmetric



Random-fair

Partial

Symmetric



Selected Reporting Strategies

Time-based

Hourly: Sensor readings are reported every hour to the application server

Daily: Sensor readings are reported once a day

Exchange-based

1-Exchange: Sensor readings are reported after each exchange

Metric-based

Jumbling-based: Sensor readings are reported when the fraction of jumbled readings reaches a given threshold (25%, 50%, or 75%)

Distance-based: Sensor readings are reported when the mean distance between the sensor readings reaches a given threshold (1 km, 2 km, or 5 km)

Evaluation: Objectives

Cross-analysis of the impact of the selected exchange and reporting strategy on the following metrics:

1. **Jumbling degree**: It measures the average percentage of reported sensor readings having been jumbled with other participants

2. **Distance**: It estimates the average distance between the actual path followed by the participants and the jumbled path resulting from the exchange

3. **Overhead**: It compares the average amount of triplets having been reported after jumbling with the amount of triplets having been collected

Evaluation: Data set

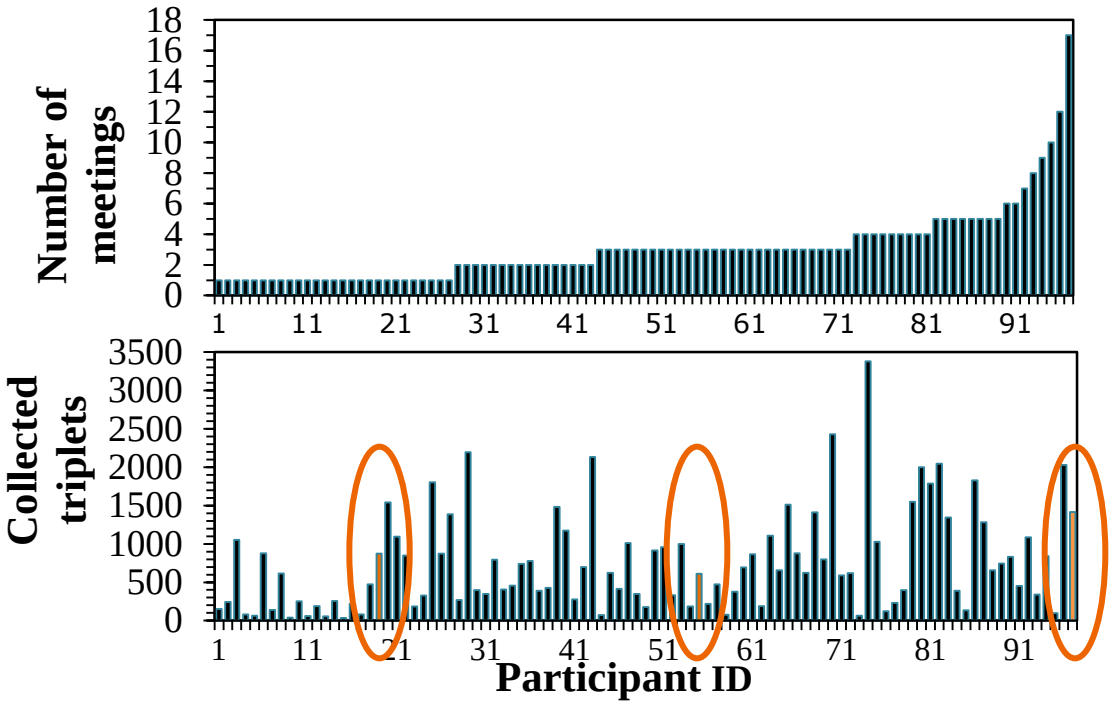
GPS traces from the GeoLife project [1]

97 participants over 24 hours with at least one meeting

Best case: 17 meetings (ID=97)

Mean case: 3 meetings (ID=55)

Worst case: 1 meeting (ID=19)



[1] GeoLife GPS Trajectories. [Online]. Available: http://research.microsoft.com/en-us/projects/geolife

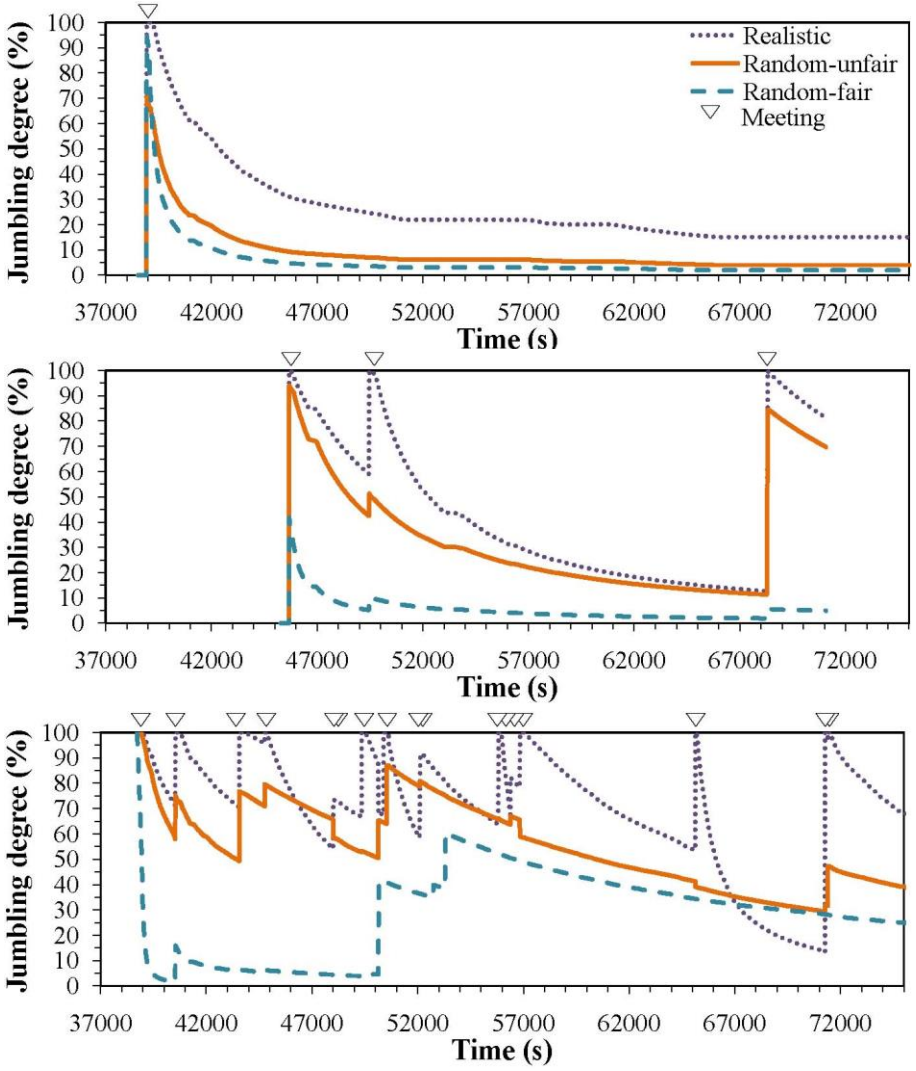
Jumbling Degree

High jumbling degree

Only few sensor readings collected by the participants themselves are reported to the application server

Little information about the participants' paths is disclosed

Provides insights about the level of obfuscation achieved at the time of the reporting to the server



Jumbling Degree

Selected results for the 97 participants

Realistic strategy:

- Jumbling degree of 100% except for time-based reporting strategies
- Paths are therefore protected independently of the selected reporting strategy
- Delivery of the sensor readings to the application after only one meeting

Random-unfair strategy:

- Jumbling degree does not reach 100% but the maxima are greater than 96%
- Best results are obtained for the jumbling-based reporting strategies

Random-fair strategy:

- Lowest jumbling degree with maxima only up to 80% due to the fairness constraint

Distance

Selected results for the 97 participants

A small distance indicates that the reported path remains in proximity of the actual path

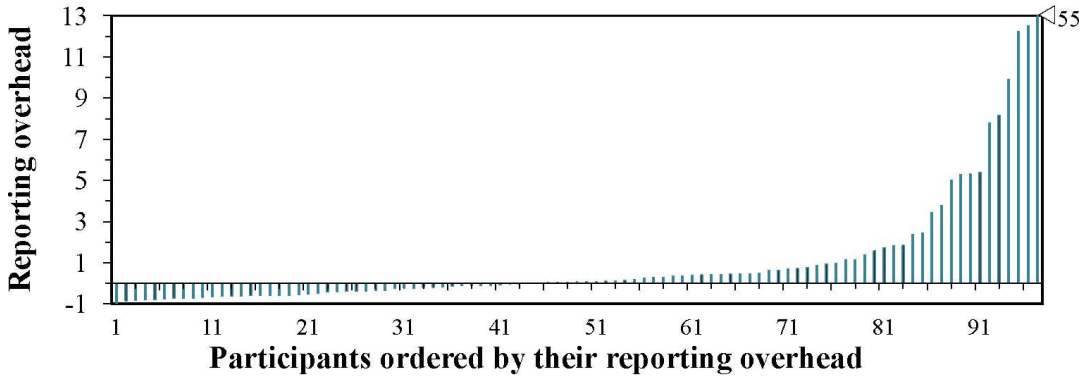
	Median distance
Realistic exchange strategy	4 km
Random-unfair exchange strategy	5 km
Random-fair exchange strategy	5 km

Exchange strategies show only slightly different distance results

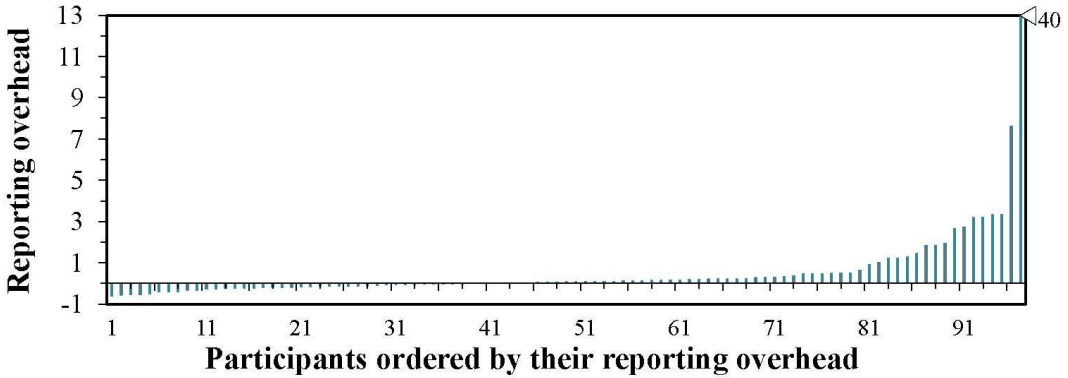
Reporting strategies have only little impact on the distance metric

Overhead

Realistic strategy



Random-unfair strategy



Summary of Results

Realistic exchange strategy

Best results in terms of jumbling degree (except for time-based reporting)

Reporting latency depends on meeting pattern, but one meeting is sufficient

Require a high degree of trust in the other participants

May introduce substantial overhead

Random-unfair

Require less trust in other participants and introduce less overhead than the realistic strategy

Performances depends on the selected random values

Additional meetings are required to provide the same guarantees as the realistic scheme

Random-fair

No reporting overhead

Roadmap

- Introduction
- Incentives
- Trust
- **Privacy**
  - Collaborative path hiding
  - **AnonySense: Anonymous Tasking and Reporting**
  - Private Data Vaults: Access Control
  - IncogniSense: Balancing Privacy and Trust
- Conclusion

AnonySense: Goals

To build an application independent infrastructure for anonymous tasking and reporting. The infrastructure enables applications to:

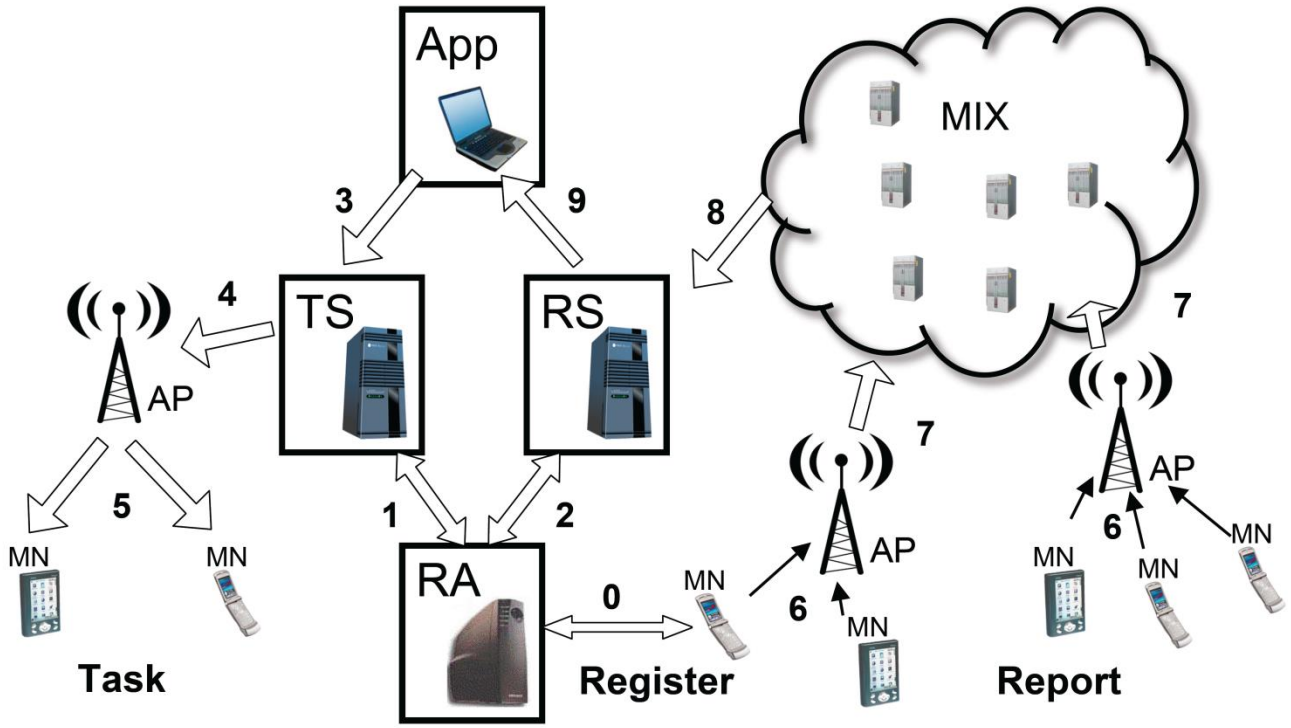
- Task a node using a new tasking language
- Anonymously distribute tasks to nodes
- Collect anonymous yet verifiable reports from nodes

Anonymity is achieved if

- No entity is able to link a report to a particular carrier
- No intermediate entity can infer information about what is reported, tamper with tasks or falsify reports

M. Shin, C. Cornelius, D. Peebles, A. Kapadia, D. Kotz and N. Triandopoulos, “AnonySense: A System for Anonymous Opportunistic Sensing” in Pervasive and Mobile Computing, May 2010.

System Architecture



TS: Task Server receives tasks descriptions from App and distributed to MNs

RS: Report Server collects and aggregates reports (for privacy) from MNs and forwards to App

RA: Registration Authority registers MNs and issues certificates to TS and RS

MIX: MIX network is the anonymisation channel. MNs send multiply encrypted messages that are “peeled off” by a layer at a time by subsequent mix nodes.

Registration

RA verifies that correct software is running on MN by leveraging software attestation (E.g. by using a TPM)

RA verifies attributes of the MN carrier

RA installs a private group key used for signing report. This is used for the group signature protocol

D. Boneh, X. Boyen and H. Schacham, “Short Group Signatures” in Proceedings of Crypto, 2004.

Tasking

A task submitted by an application to the task server is first evaluated by the RA

- Each submitted task has certain attributes
- The RA makes sure that there are at least k >= k_g nodes that can satisfy the criteria, where k_g is a global parameter
- This prevents against targeting a narrow subset of users

MNs then poll the task server at random intervals using recycled IP addresses to get a task list

TS verifies that MN is valid by providing a nonce challenge. MN replies by signing the nonce with the group key.

Reporting

A report submitted by a MN first goes through the MIX network

The MIX network ensures that when the reports arrive at the RS, they are mixed with reports from other users

The MIX network can delay messages till they can be reliably mixed

The RS adds another layer of privacy by providing k-anonymity

MNs recycle MAC address before submitting reports

Threat Model

Eavesdropping is prevented because of encrypted communications

Adversary cannot pose as a TS or RS because each have certificates

TS cannot link tasks to users because of recycled IP and MAC addresses and random polling

Adversary cannot learn much by submitted tasks to the system since a valid task must be executable by $k >= k_g$ mobile nodes

Adversarial MN may receive tasks but cannot see them because

- TS validates MN before providing tasks
- RA certifies MN has a TPM before registering it
- Software never divulges tasks

Standard techniques are used to prevent replays (e.g. using nonces), message tampering (hash) and non repudiation (digital signatures)

Roadmap

- Introduction
- Incentives
- Trust
- **Privacy**
  - Collaborative path hiding
  - AnonySense: Anonymous Tasking and Reporting
  - **Private Data Vaults: Access Control**
  - IncogniSense: Balancing Privacy and Trust
- Conclusion

Access Control

Data gathered through participatory sensing is personal as well as being valuable

- it quantifies habits, routines, associations and is easy to mine

Users should have a greater control and say over who has access to their data

An architecture is necessary that can

- Protect individual privacy
- Document ownership
- Provide visibility of processing

Personal Data Vaults

Individually-controlled secure data repositories

Separation of data collection and archiving from sharing

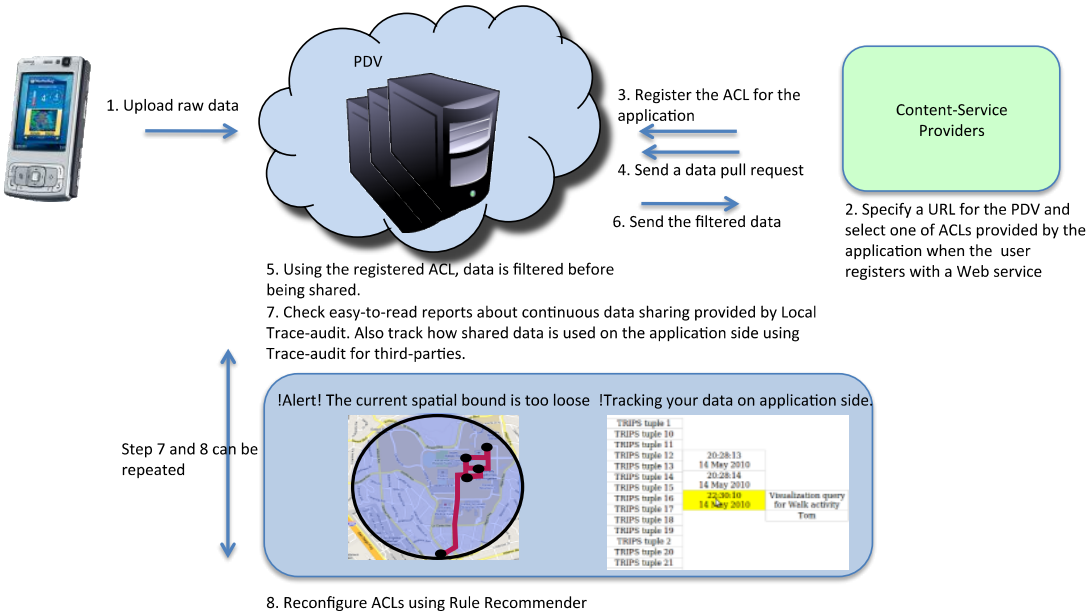
No reliance on third-parties to control data sharing

Key system components:

- Granular Access Control Lists (ACL): who has access to data and at what resolution
- Trace-audit: logs transactions and provide users with visual representations of who accessed their data and how
- Rule recommender: pre-calculated constraints for pre- determined privacy policies

M. Mun, N. Mishra, K. Shilton, J. Burke, D. Estrin, M. Hansen, R. Govindan, “Personal Data Vaults: A Locus of Control for Personal Data Storage”, in Proceedings of ACM CoNEXT, December 2010

PDV: Usage Scenario



Granular ACL

Entity: type and name of third parties accessing the data

Filters: constraints (type, attribute) which define data to be shared

Types and attributes of ACL constraints

Constraint	Type	Attributes
Bound	time	starttime, endtime
	location	format(in-circle,out-circle), center(GPS coordinates), radius(in km)
	number	lower, lowersymbol(=, <,<=), upper, uppersymbol(=,>,>=)
	text	attrname, text, symbol(=,!=)
Precision	time	value(private, second, minute, hour)
	location	value(private, exact, street, zipcode, state, country)
	number	value(private, average), timeframe(mintue, hour, day, week, month)
Frequency	time	unit(second, minute), value

ACL Example

Rule Implication	If the application named Ambulation queries, share the exact location when the user's in Westwood (within 1.5 km of the GPS coordinates of (34.06,-118.44)), otherwise, share location at a zip code level
ACL Representation	{"entity": {"type":"application","name":"ambulation"},"filters": [{"bound": [{"type":"location","format":"in-circle","center": {"latitude":34.06,"longitude":-118.44} ,"radius":1.5}],"precision": [{"type":"location","value":"exact"}] }, {"bound": [{"type": "location","format":"out-circle","center":{"latitude":34.06,"longitude":-118.44}, "radius":1.5}] , "precision": [{"type":"location","value":"zipcode"}] }] }

Trace Audit

local trace-audit: log operations performed inside PDV

third-party trace-audit: log operations that take place on the data in third-party apps

log presented to user to interpret what data has been shared with which apps

E.g.

<timestamp:2010-05-14 20:28:14, userId:System, opType:Data read for speed calculation, dataTable:GPS RAW, tupleRange:[start:10500, end:11000]>, <timestamp:2010-05-14 20:28:14, userId: System, opType:Speed values added, dataTable:TRIPS, tupleRange:[start:2300, end:2380]>

Rule Recommender

High-level interface for setting sharing policies

Pre-computed ACLs (from historical data) for high- level user interactions

Computing Bounds

- Identification of significant and routine locations + spatial bounds

Computing Precision

- Location (aggregation) tree based on significant locations

Computing Frequency (Sampling)

- Significant location identification rate calculation

Roadmap

- Introduction
- Incentives
- Trust
- **Privacy**
  - Collaborative path hiding
  - AnonySense: Anonymous Tasking and Reporting
  - Private Data Vaults: Access Control
  - **IncogniSense: Balancing Privacy and Trust**
- Conclusion

Balancing Privacy and Trust

Privacy mechanisms aim to delink sensing data streams

Establishing trust requires linking of data streams

How do we resolve this conflict?

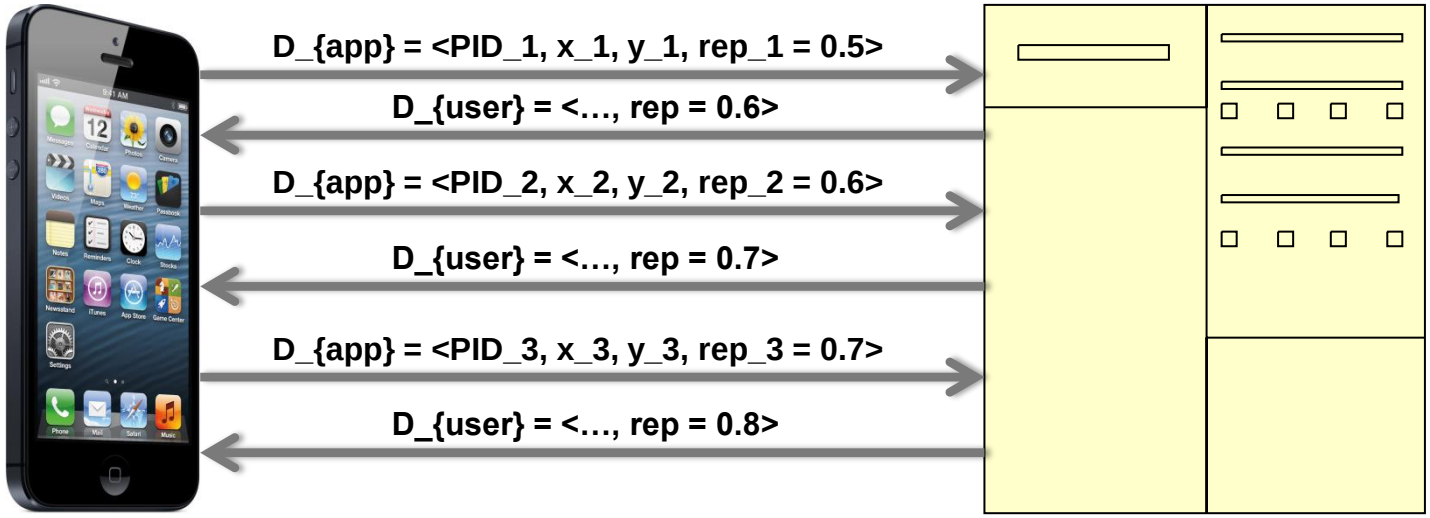
Unlink contributions (Privacy)

vs

Link contributions (Reputation)

D. Christin, C. Rosskopf, M. Hollick, L. A. Martucci and S. Kanhere, “IncogniSense: An Anonymity-preserving Framework for Participatory Sensing Applications, in IEEE PerCom 2012.

How reputation breaks privacy



	t = 1	t = 2	t = 3
pseu_1	0.5 | 0.6		
pseu_2		0.6 | 0.7	
pseu_3			0.7 | 0.8

IncogniSense: Principles

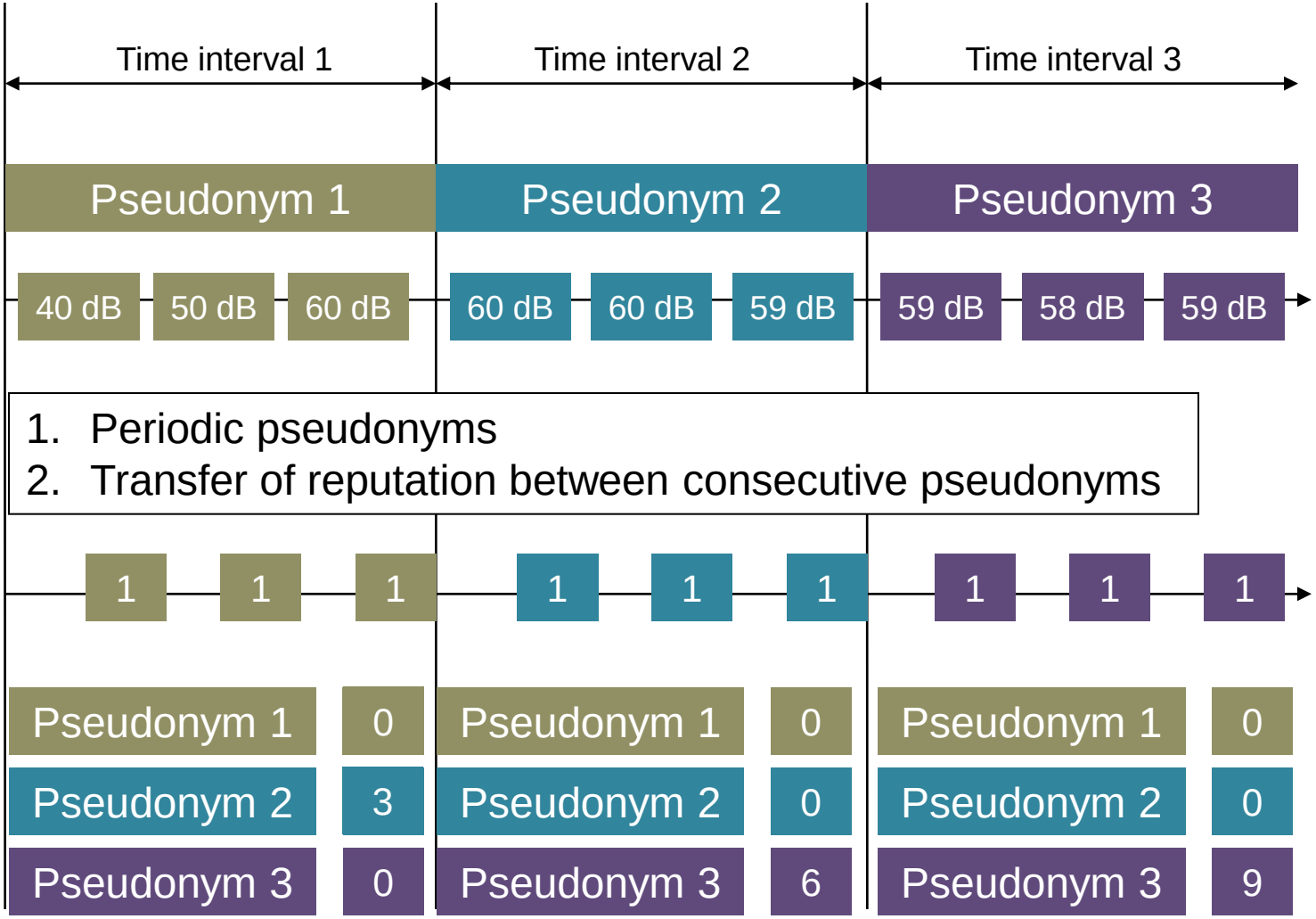
Mobile phone

Sensor readings

Application server

Reputation scores

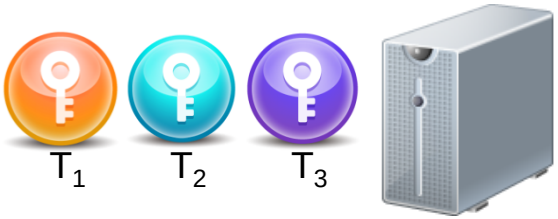
Reputation accounts



Generation of Psuedonyms

Application server

Reputation & Pseudonym Manager (RPM)



1

- Verifies pseudonym
- Blindly signs the pseudonym

- U Pseudonym 1 ount

Pseudonym 1 1

Client

Pseudonym 1

- Achieves pseudonym creation

Pseudonym 1

40 dB

- Collect sensor readings

1. Ensures authenticity of pseudonym without revealing it to the RPM
2. RPM cannot link the pseudonym with the ID of its creator
3. RPM guarantees that each client has a unique pseudonym

Generation of Reputation Tokens

RPM


RT value=1


RT value=2


RT value=3

Pseudonym 1	0
Pseudonym 2	3

Client

Current pseudonym:

Pseudonym 2

- Creates new pseudonym


Pseudonym 2

- Requests its reputation score at RPM
- Prepares reputation token for signature
- Ac  es

RT

RT

- Verifies the reputation
- Blindly signs the reputation token

RT

- Verifies RT
- Invalidates RT

1. Ensures authenticity of RT without revealing it to the RPM
2. RPM cannot link the RT with the current pseudonym
3. RPM guarantees that each client does not abuse the system

Reputation Linking Attack

1. Ensures authenticity of pseudonym without revealing it to the RPM
2. RPM cannot link the pseudonym with the ID of its creator
3. RPM guarantees that each client has a unique pseudonym

1. Ensures authenticity of RT without revealing it to the RPM
2. RPM cannot link the RT with the current pseudonym
3. RPM guarantees that each client does not abuse the system

RPM

Pseudonym A	10
Pseudonym B	2
Pseudonym C	0

Time interval 1

RPM

Pseudonym A	0
Pseudonym B	0
Pseudonym C	0
Pseudonym D	12
Pseudonym E	4
Pseudonym F	2

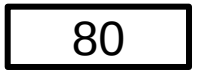
Time interval 2

Proposed Reputation Cloaking Mechanisms

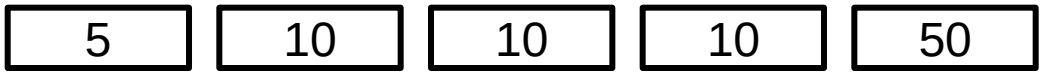
Assume a reputation score value 85 to be transferred.

- **Full scheme:**

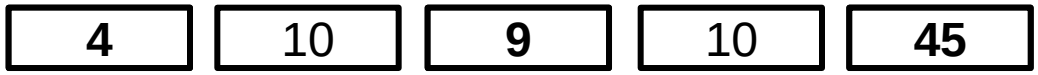
- **Floor scheme** (e.g., [70;79], [80;89]):



- **RandSet scheme** (e.g., (5,10, 50)):



- **RandScore scheme:**



- **Hybrid scheme**

Evaluation: Objectives

1. Measure the **level of anonymity** of the proposed cloaking schemes

2. Quantify the **reduction in reputation score** caused by the cloaking schemes

3. Measure the **overhead** in terms of energy consumption for the clients under real-world conditions

Simulation Settings

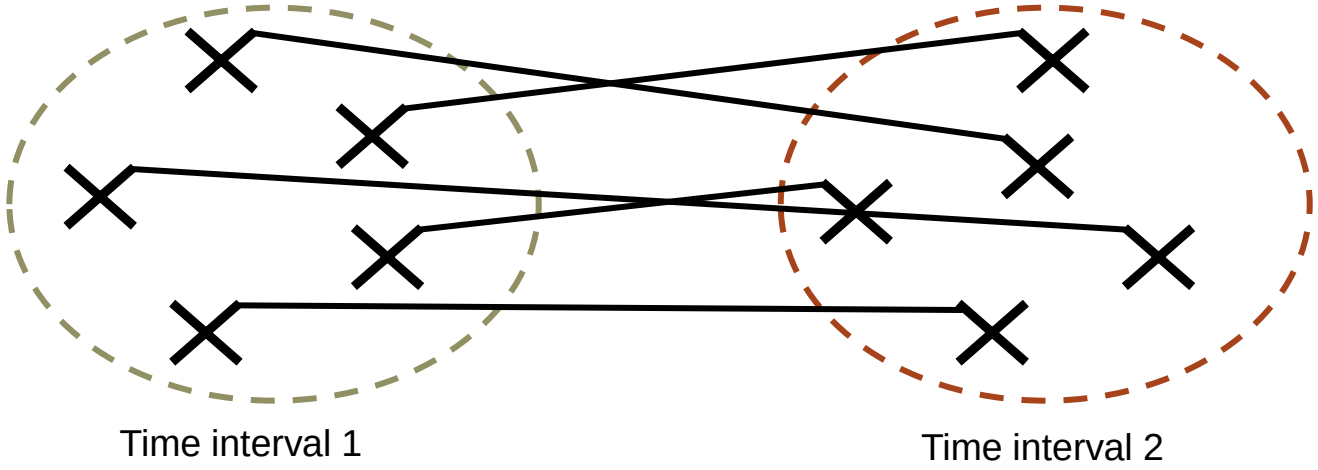
100 time intervals (T) and 100 simulation runs

100 **continuously active** clients reporting 5 sensor readings per interval

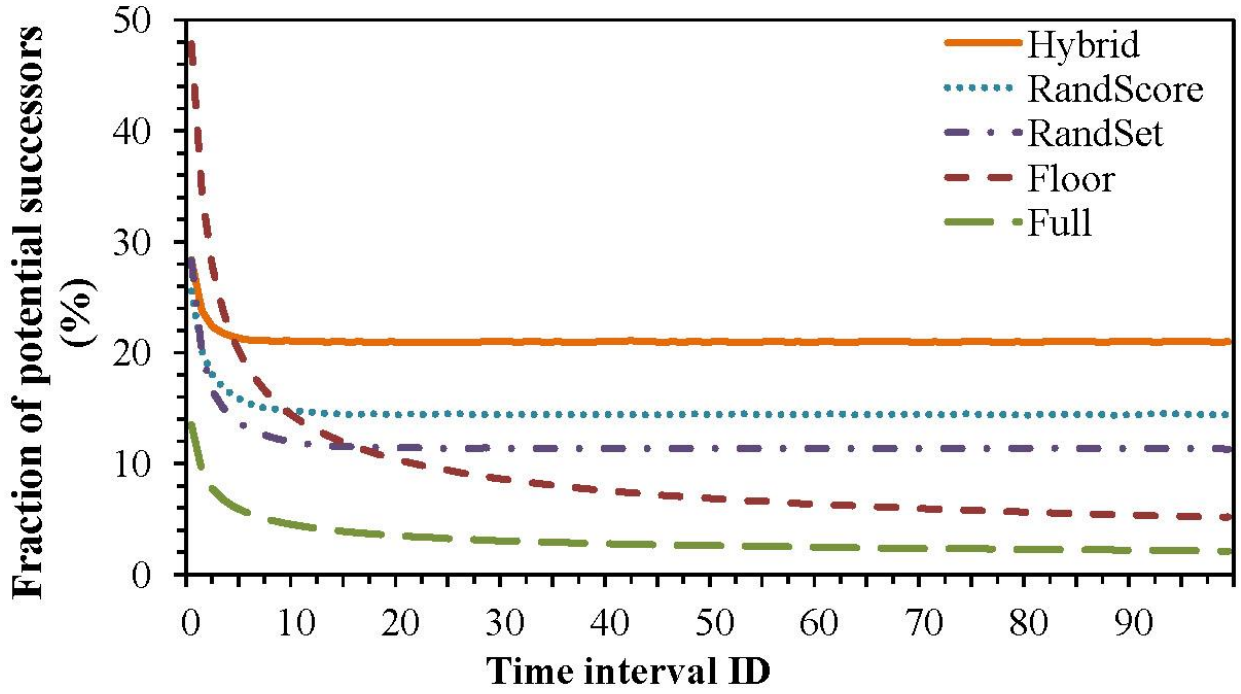
Application runs a simulated reputation algorithm

RPM and application server are **malicious internal observers**

- **Link consecutive pseudonyms based on the transferred reputation**
- Use bijection between sets of pseudonym active in subsequent intervals

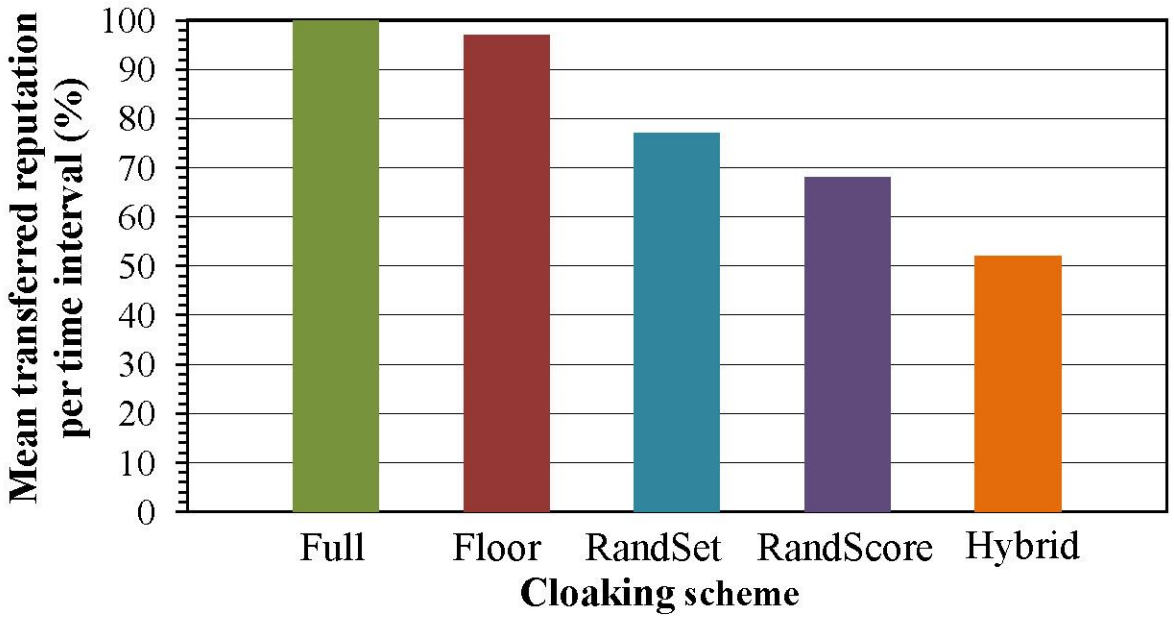


Level of Unlikability



Floor scheme: The interval size is equal to 20
RandSet scheme: The probability to discard RTs is set to 20%
RandScore scheme: The probability to lower a RT value is set to 25%

Loss in Reputation



Level of unlinkability: Full < Floor < RandSet < RandScore < Hydrid

SUMMARY
&
CONCLUSION

Summary

Incentives

- Mechanism design
- All-pay auctions
  - Heterogeneous
- Tullock contests

Trust

- Reputation based
- Social-network based

Summary

Privacy

- **Location privacy**
- **Anonymous tasking and reporting**
- **Access control of personal data**
- **Balancing trust and privacy**

Research directions

Incentives

- Bounded rationality
- Correlation among beliefs
- Collusion resistant

Trust

- Data quality validation
- Peer assessment
- Collusive & Sybil attacks

Privacy

- Behavioral privacy leakage
- Better access control and storage
- Privacy in the face of big data

References (and references therein)

- Vijay Krishna, *Auction Theory*, 1st (2002) and 2nd (2009) editions. Academic press.
- Y. Narahari (2014), Game Theory and Mechanism Design. IISc Press and WSPC.
- Nisan, Roughgarden, Tardos and Vazirani (2007), Algorithmic Game Theory. Cambridge University Press.
- Milan Vojnović, Mechanism design. Microsoft Research.
- T. Luo, S. K. Das, H-P. Tan, and L. Xia, “Incentive mechanism design for crowdsourcing: an all-pay auction approach”, ACM TIST, vol. 7, no. 3, pp. 35:1-26, 2016.
- T. Luo, S. S. Kanhere, S. K. Das, and H-P. Tan, “Incentive mechanism design for heterogeneous crowdsourcing using all-pay contests”, IEEE Transactions on Mobile Computing (TMC), 2016.
- T. Luo, H-P. Tan, and L. Xia, “Profit-Maximizing Incentive for participatory sensing”, IEEE INFOCOM, 2014.
- T. Luo, S. S. Kanhere, S. Das, and H-P. Tan, “Optimal Prizes for All-Pay Contests in Heterogeneous Crowdsourcing”, IEEE MASS, 2014.
- T. Luo, S. S. Kanhere, H-P. Tan, F. Wu, and H. Wu, “Crowdsourcing with Tullock Contests: A New Perspective”, IEEE INFOCOM, 2015.
- K. Huang, S. S. Kanhere and W. Hu, On the Need For a Reputation System in Mobile Phone Based Sensing, in Ad Hoc Networks, vol. 12, pp. 130-149, January 2014.

References (contd.)

- T. Luo, S. S. Kanhere, and H-P. Tan, “SEW-ing a Simple Endorsement Web to incentivize trustworthy participatory sensing”, IEEE SECON, 2014.
- D. Christin, J. Guillemet, A. Reinhardt, M. Hollick, S. S. Kanhere, “Privacy Preserving Collaborative Path Hiding for Participatory Sensing Applications”, in Proceedings of IEEE MASS 2011, Valencia, Spain, October 2011.
- M. Shin, C. Cornelius, D. Peebles, A. Kapadia, D. Kotz and N. Triandopoulos, “AnonySense: A System for Anonymous Opportunistic Sensing” in Pervasive and Mobile Computing, May 2010.
- M. Mun, N. Mishra, K. Shilton, J. Burke, D. Estrin, M. Hansen, R. Govindan, “Personal Data Vaults: A Locus of Control for Personal Data Storage”, in Proceedings of ACM CoNEXT, December 2010
- D. Christin, C. Rosskopf, M. Hollick, L. A. Martucci and S. Kanhere, “IncogniSense: An Anonymity-preserving Framework for Participatory Sensing Applications, in IEEE PerCom 2012.