

Distributed Data Mining in Peer-to-Peer Networks: Local Algorithms, Privacy Issues, and Games

Hillol Kargupta

University of Maryland, Baltimore County and AGNIK

www.cs.umbc.edu/~hillol

Acknowledgement:

Kun Liu, Kamalika Das, Ran Wolff, Kanishka Bhaduri

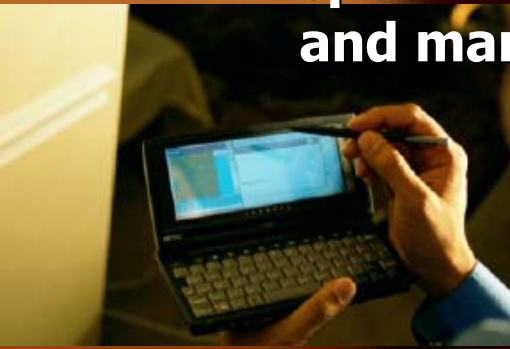
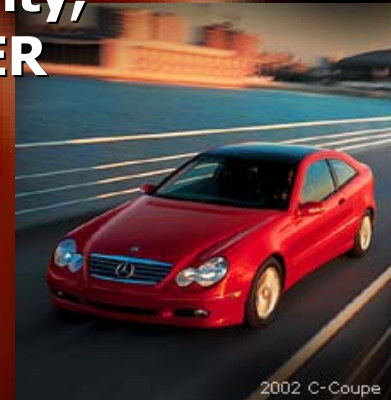
US National Science Foundation

Roadmap

- Introduction
 - Distributed Data Mining and Peer-to-Peer Networks
 - Local Algorithms
 - Exact Local Algorithms
 - Approximate Local Algorithms
 - Privacy Issues
 - Assumptions and Problems
 - A Game Theoretic Perspective
 - An Illustration using Multi-party Secure Sum
 - Conclusions
-

Research & Development at UMBC DIADIC Laboratory and AGNIK, LLC

- **Distributed and mobile data mining.**
- **Supported by Department of Homeland Security, NASA, US National Science Foundation CAREER award and other grants, US Air Force, TRW Research Foundation, Maryland Technology Development Council, and others.**
- **Agnik, LLC: A Spin-off from DIADIC Lab, specializing on mobile and distributed data mining and management.**



Data Mining and Distributed Data Mining (DDM)

- Data Mining: Scalable analysis of data by paying careful attention to the resources:
 - computing,
 - communication,
 - storage, and
 - human-computer interaction.
- Distributed data mining (DDM): Mining data using distributed resources.

Distributed Data Mining: Application Domains

- ❑ **Mining Large Databases from distributed sites**
 - Grid data mining in Earth Science, Astronomy, Counter-terrorism, Bioinformatics
 - ❑ **Monitoring Multiple time critical data streams**
 - Monitoring vehicle data streams in real-time
 - Monitoring physiological data streams
 - ❑ **Analyzing data in Lightweight Sensor Networks and Mobile devices**
 - Limited network bandwidth
 - Limited power supply
 - ❑ **Preserving privacy**
 - Security/Safety related applications
 - ❑ **Peer-to-peer data mining**
 - Large decentralized asynchronous environments
-

What is a Peer-to-peer (P2P) Network?

- Relies primarily on the computing resources of the participants in the network rather than a relatively low number of servers.
 - P2P networks are typically used for connecting nodes via largely ad hoc connections.
 - No central administrator/coordinator
 - Peers simultaneously function as both "clients" and "servers"
 - Privacy is an important issue in most P2P applications
-

Where do we find P2P Networks?

- File-sharing networks
 - KaZAa, Napster, Gnutella
- Sensor Networks?
- Mobile Ad-hoc NETWORK (MANET)?

- Next Generation:
 - P2P Search Engines
 - Social Networks
 - Digital libraries
 - P2P “YouTube”?



Motivation : P2P Search Engine

What is the most visited news-page in network today?

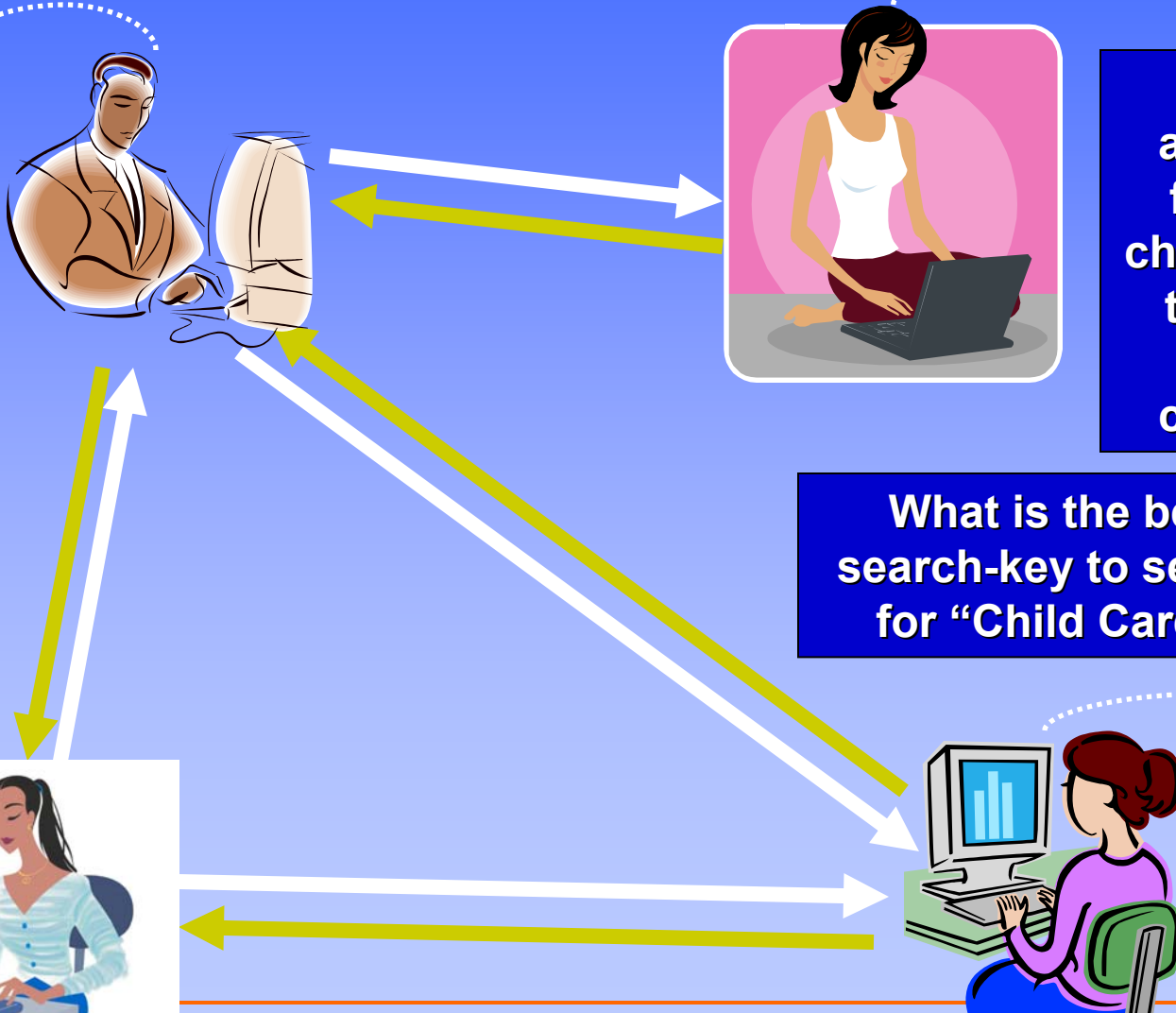


Has anybody found a cheap store to buy a digital camera?

What is the most-likely browsing pattern to know about "Data Mining"?



What is the best search-key to search for "Child Care"?



Data Sources

- Web-browser history
- Browser cache
- Click-stream data stored at browser (browsing pattern)
- Search queries typed in the search engine
- User profile
- Bookmarks

Challenges

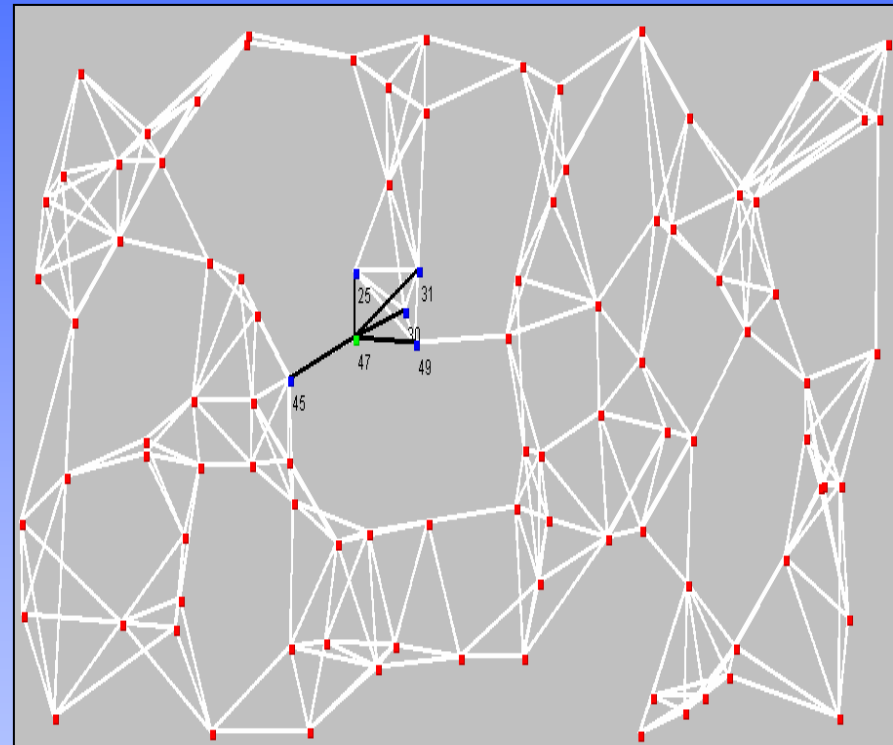
- Indexing, clustering, data analysis in a decentralized asynchronous manner
 - Scalability
 - Privacy
-

Related Work on Peer-to-Peer Data Mining

- Distributed Majority Voting Algorithm (Gifford, 1979; Thomas, 1979; Wolff, Schuster, 2003)
 - P2P Association Rule Learning (Wolff, Schuster, 2003)
 - Gossiping (Kempe, Dobra, Gehrke 2003)
 - Bandhopadhaya et al.'05
 - P2P L2 Norm Monitoring (Wolff, Bhaduri, Kargupta, 2005)
 - P2P Clustering (Dutta, Giannella, Kargupta, 2006)
 - P2P Random Sampling (Dutta, Gianella, Kargupta, 2007)
-

Locality Sensitive Distributed Algorithms

- Global algorithms: Know everything about the entire network
 - Every node needs to maintain information about the entire network
 - Maintaining this information is resource intensive for large networks
- Local algorithms: Communicate only with the local neighborhood.
- Does locality imply efficiency?



Bounded Communication Local Algorithms

- Every node communicates with its local neighborhood
 - In addition, the total amount of communication with its neighbors is also bounded
-

Defining the Problem

- Let $G=(V, E)$ be a graph
- Let Ω_k be the set of all neighboring nodes of the k -th node $v_k \in V$

- Need a decomposable representation where $f(V)$ can be computed from locally computed functions $\Phi_k(\Omega_k)$

- Example:

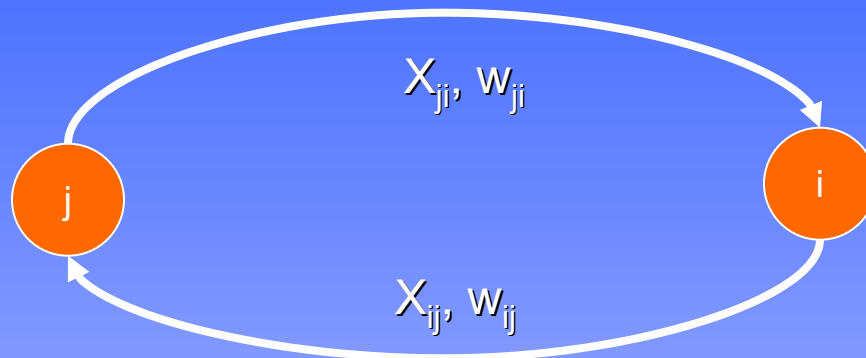
$$f(V) = \sum_k w_k \Phi_k(\Omega_k)$$

Approaches

- Function computation through decomposable representations
 - Exact decompositions
 - Deterministic techniques
 - Approximations
 - Randomized techniques
 - Sampling-based approximations
 - Variational approximations
-

Local L2 Norm Monitoring Algorithm

- Initial setup: each peer has
 - A data vector
 - Some global pattern vector

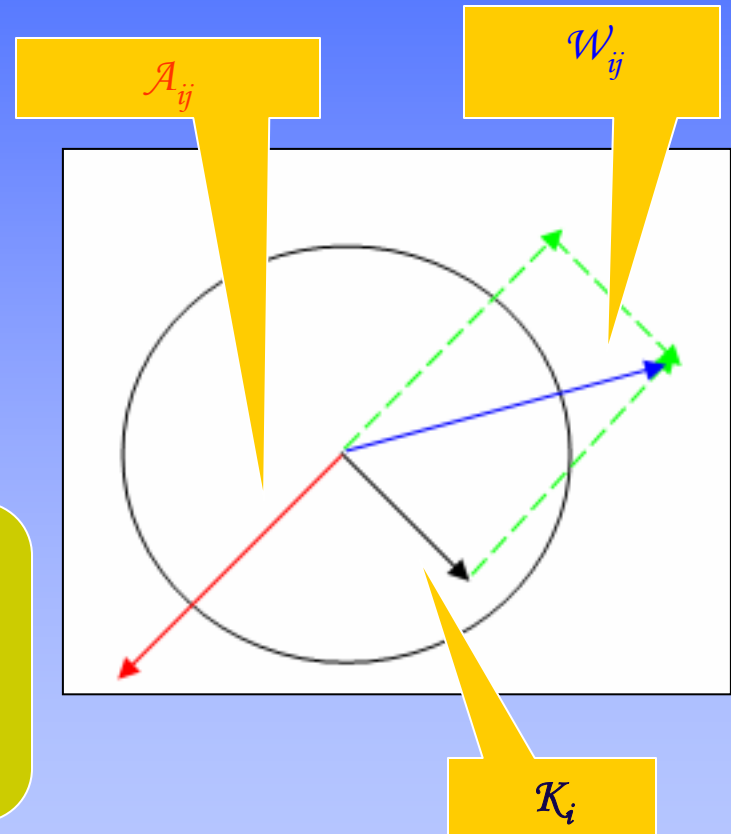


- Monitoring Problem:
 - Is the L2 norm of the distance between the average data vector and the pattern vector greater than a given constant ϵ
 - Applications:
 - Centroid monitoring
 - Eigenvector monitoring
-

Notations

Transform the task to a geometric problem

- P_1, \dots, P_n – set of peers
- P_i 's local vectors -
 - $S_{i,t}$ – data at time t
 - X_{ij} – sent by P_i to P_j
 - \mathcal{K}_i – *knowledge* ($S_{i,t} + X_{ji}$)
 - \mathcal{A}_{ij} – **agreement** ($X_{ij} + X_{ji}$)
 - \mathcal{W}_{ij} – **withheld** ($\mathcal{K}_i - \mathcal{A}_{ij}$)
 - \mathcal{G}_t – average of all peers



All vectors computations are local to a peer

Possibilities

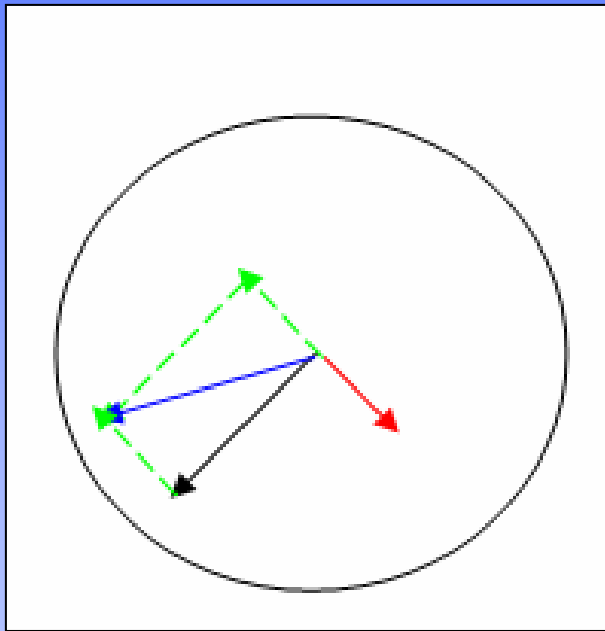
1. All 3 vectors inside circle
2. All 3 vectors outside circle
3. Some are inside, some are outside

Theorem:

If for every peer and each of its neighbours both the agreement and the withheld knowledge are in a convex shape (here a circle) - then so is the global average

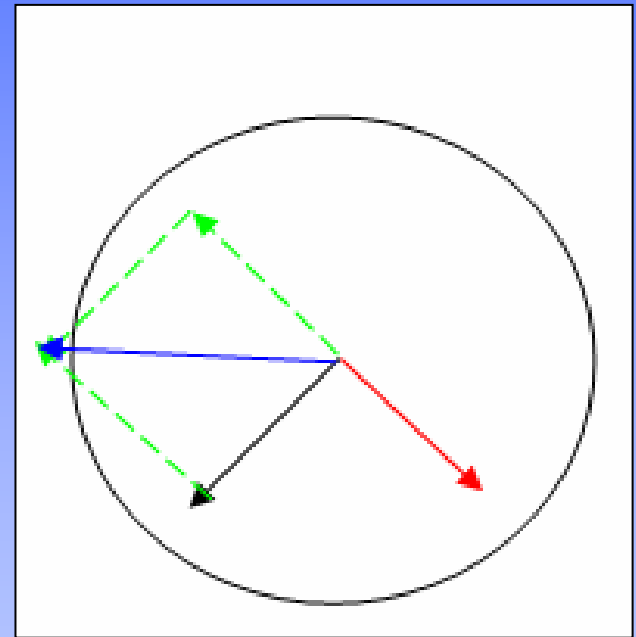
[Wolff, Bhaduri, Kargupta, 2006]

Case 1 : All Inside Circle



No more communication

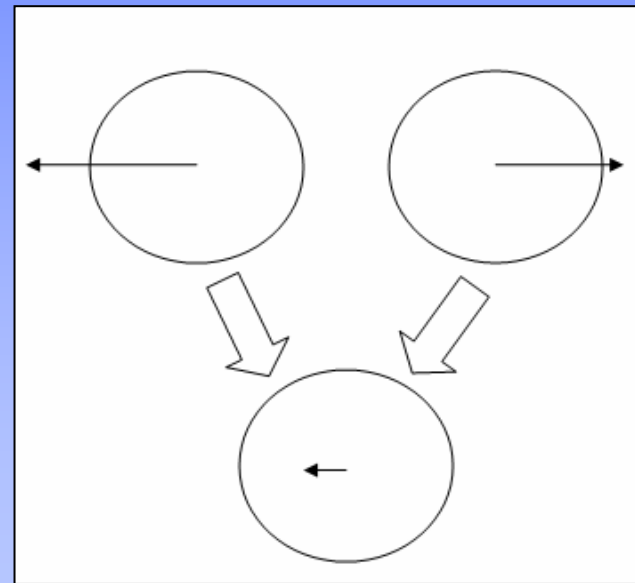
Case 3 : Inside & Outside



Needs communication

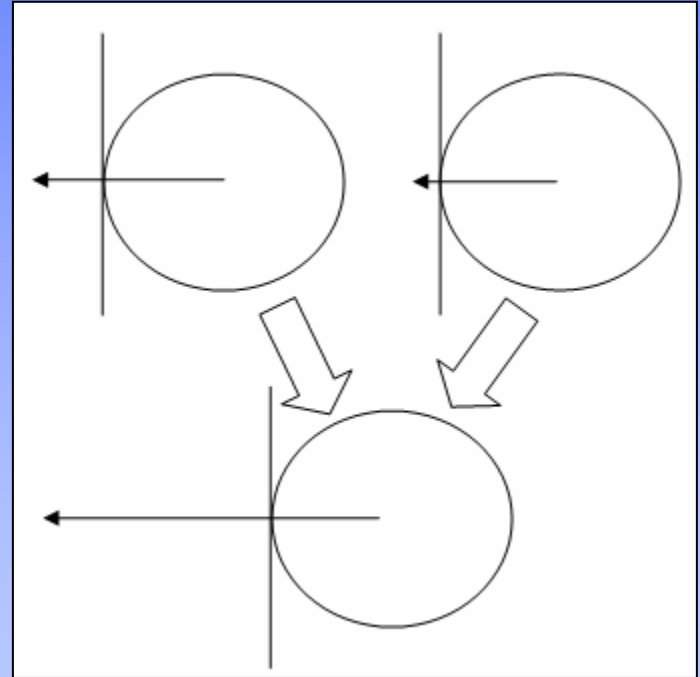
Case 2: All Outside Circle

- Two peers independently estimate that global average vector outside
- Combined average can still be inside !!!

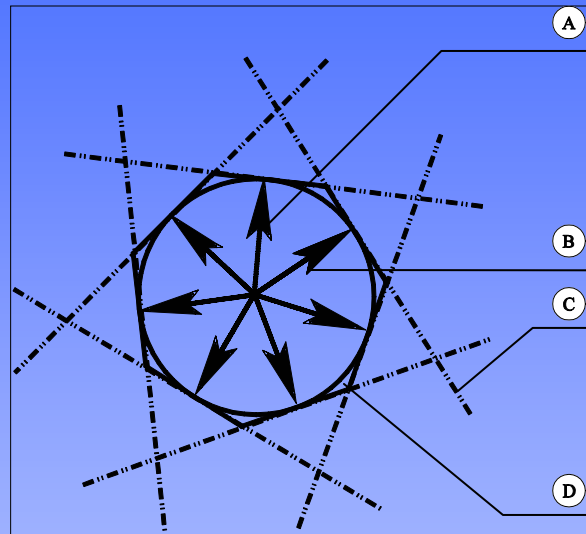


Case 2: All Outside Circle

- Solution – use tangent lines to bound circle
- A tangent or half-space is itself an unbounded convex region
- The theorem holds in this case as well



Overall Algorithm



(A) Area inside ε circle. (B) Seven evenly spaced vectors. (C) Borders of seven half-spaces $u_i \cdot x \geq \varepsilon$ define a polygon. (D) Area between circle and union of half-spaces

References

- R. Wolff, K. Bhaduri, H. Kargupta. (2006). Local L2 Thresholding Based Data Mining in Peer-to-Peer Systems. Proceedings of the 2006 SIAM International Data Mining Conference, pp. 430-441.
- S. Datta, K. Bhaduri, C. Giannella, R. Wolff, H. Kargupta. (2006). Distributed Data Mining in Peer-to-Peer Networks. IEEE Internet Computing special issue on Distributed Data Mining. volume 10, number 4, pages 18--26.
- R. Wolff, K. Bhaduri, H. Kargupta. (2006). A Peer-to-Peer Distributed Algorithm for Eigenstate Monitoring. Data Mining and Knowledge Discovery Journal.

Extension: Computing Cluster Centroid

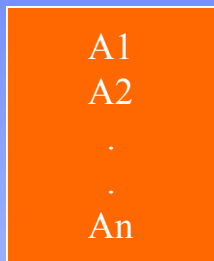
- Beyond Monitoring
 - Exact *local* algorithm not available
 - How about Approximation?
-

Approximation

- Estimate $\Phi_k(\Omega_k)$
 - Cardinal sampling
 - Ordinal relaxation
 - Interested in constructing an ordering
 - Find the ones that rank high

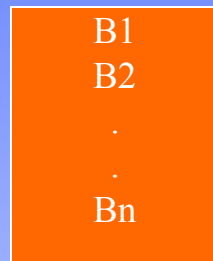
Cardinal Approximation: Inner Product

Node 1



$Z_{1,k}$

Node2

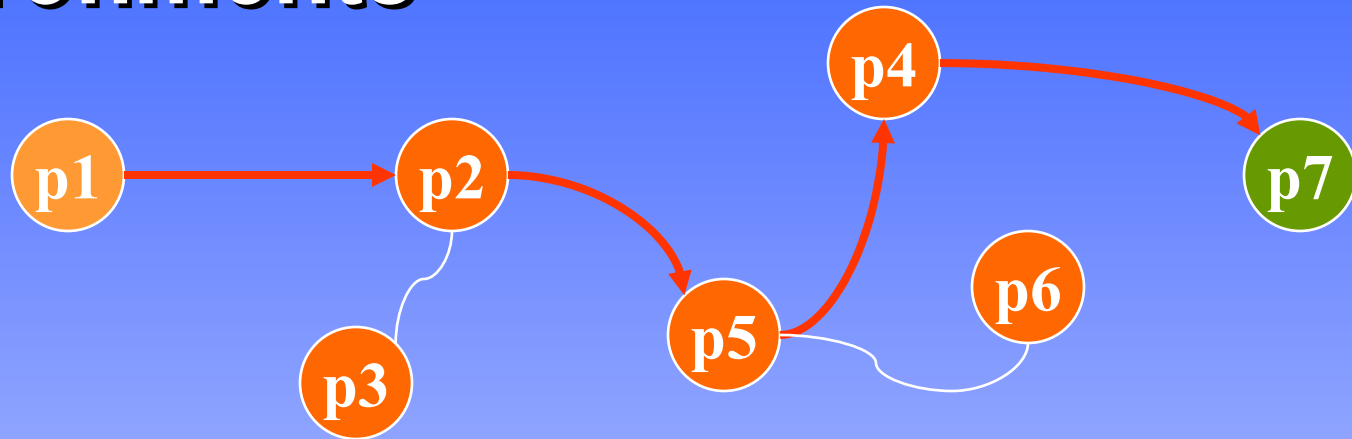


$Z_{2,k}$



- Node 1 computes $Z_{1,k}$
 - $Z_{1k}=A1.J_1+..+An.J_n$
 - $J_i \in \{+1,-1\}$ with uniform probability
- Node 2 calculates $Z_{2,k}$
 - $Z_{2k}=B1.J_1+..+Bn.J_n$
- Compute $z_{1,k} \cdot z_{2,k}$ for a few times and take the average

Sampling in Distributed Environments



- i.i.d sampling through random walk: Needs more attention for data mining application
- Metropolis-Hasting algorithm for random walk with $O(\lg n)$ steps for selecting i.i.d. samples
- Issues:
 - Nodes may have different degrees
 - Nodes may contain different number of data tuples

Random Sampling

Protocol 5.2.1 Metropolis-Hastings Random Walk

- 1: **FOR** each node i , $1 \leq i \leq n$
 - 2: **IF** receives a query q
 - 3: Replies with d_i
 - 4: **IF** receives a random walk message
 - 5: **IF** TTL == 0
 - 6: Terminates the walk
 - 7: **ELSE**
 - 8: TTL = TTL - 1
 - 9: Sends out a query q to its neighbors $\Psi(i)$
 - 10: **IF** receives all the replies from its $\Psi(i)$
 - 11: Modifies transition probability p_{ij} as follows:
 - 12:
$$p_{ij} = \begin{cases} 1 / \max(d_i, d_j) & \text{if } i \neq j \text{ and } j \in \Psi(i) \\ 1 - \sum_{k \in \Psi(i)} p_{ik} & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$$
 - 13: Walk to next node with probability p_{ij} .
-

Ordinal Relaxation

- Let X be a continuous random variable
- Let ξ_p be the population percentile of order p , i.e. $\Pr\{x \leq \xi_p\} = p$
- Let $x_1 < x_2 < \dots < x_N$ be N independent samples from X
- We have

$$\Pr\{x_N > \xi_p\} > q \Rightarrow N \geq \left\lceil \frac{\log(1 - q)}{\log p} \right\rceil$$

- Example:
 - $q=95\%$ and $p=80\% \rightarrow N=14$
 - If we took 14 independent samples from any distribution, we can be 95% confident that 80% of the population would be below x_{14} .

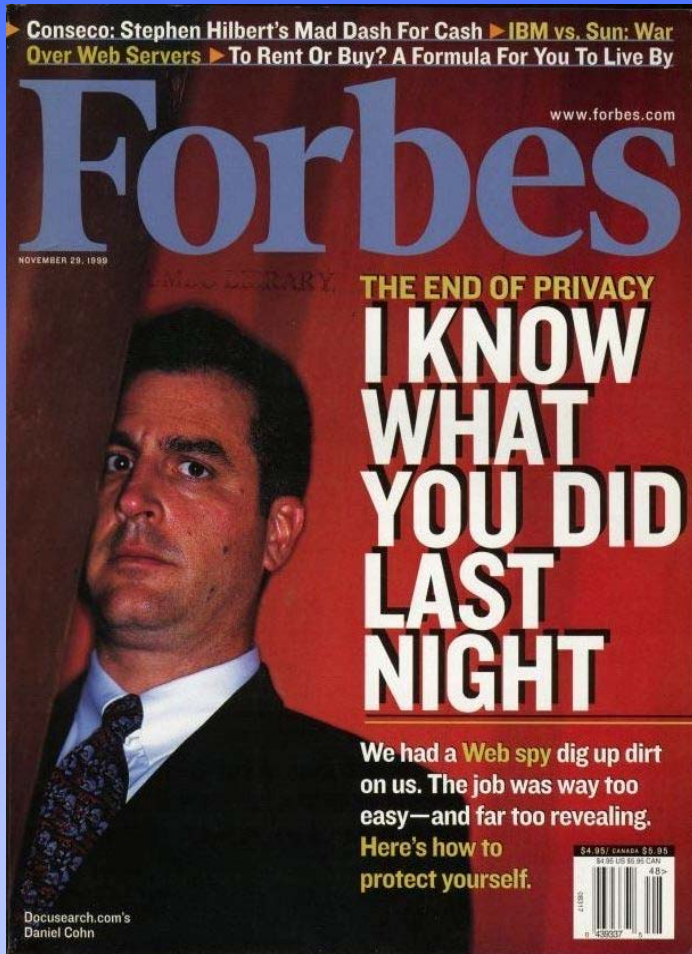
Ordinal Inner Product Computation

- Each node has a vector X_i
- Compute the Inner Product Matrix
 - Every node needs X_j from every node.
- How about finding just the top-k entries of the inner product matrix?

References

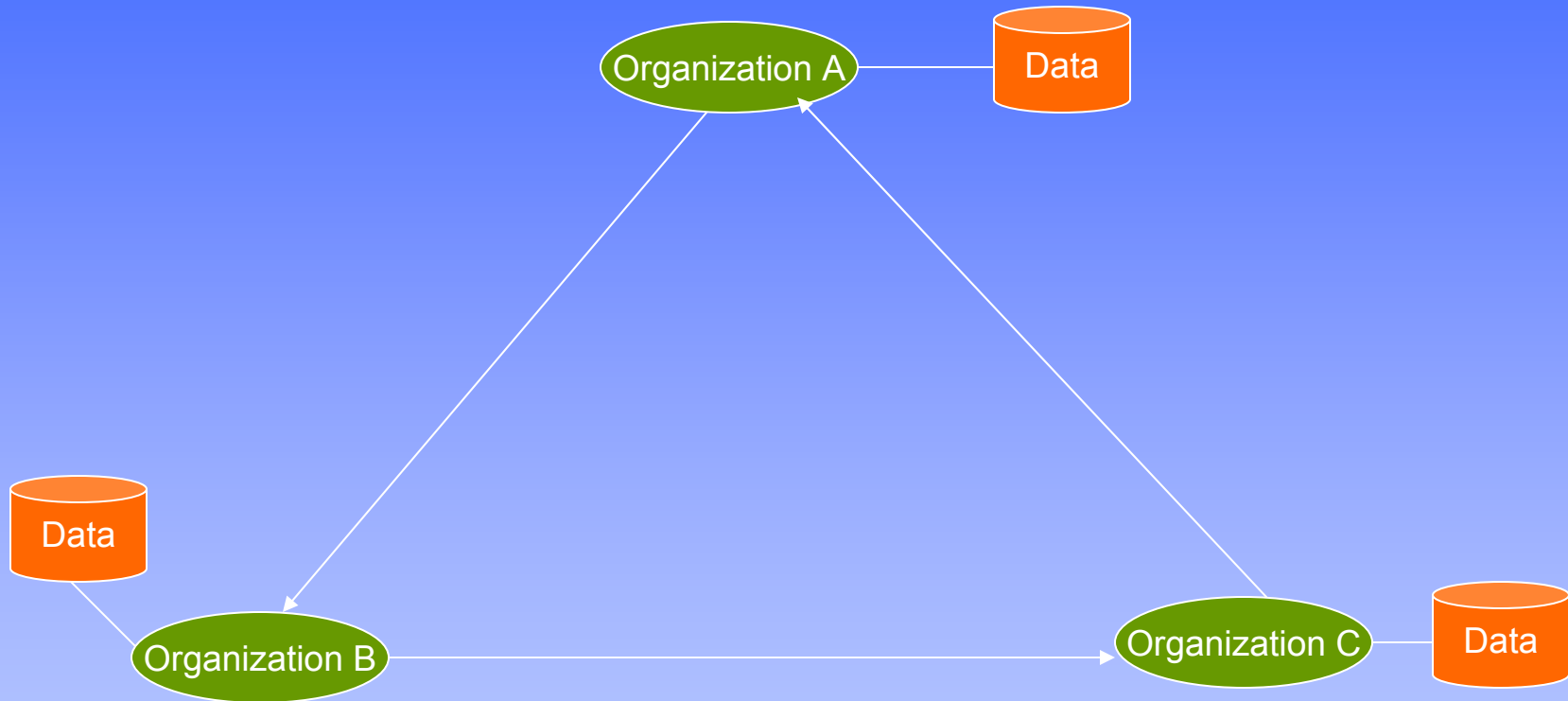
- K. Das, K. Bhaduri, and H. Kargupta. (2006). An Ordinal Framework for Identifying Significant Inner Product Elements in a Peer-to-Peer Network. IEEE Transactions on Knowledge and Data Engineering.
 - J. Branch, B. Szymanski, R. Wolff, C. Gianella, H. Kargupta. (2006). In-Network Outlier Detection in Wireless Sensor Networks. Proceedings of the 26th International Conference on Distributed Systems, 2006.
 - S. Datta, C. Giannella, H. Kargupta. (2006). K-Means Clustering over a Large, Dynamic Network. Proceedings of the 2006 SIAM International Data Mining Conference, pp. 153-164.
-

Privacy and Data Mining



- ┌ “the best (and perhaps only) way to overcome the ‘limitations’ of data mining techniques is to do more research in data mining, including areas like data security and privacy-preserving data mining, which are actually active and growing research areas.”
- SIGKDD Executive Committee, “‘Data Mining’ Is NOT Against Civil Liberties,” 2003.
- ┌ Privacy-preserving data mining is “the study of how to produce valid mining models and patterns without disclosing private information.”
- F. Giannotti and F. Bonchi, “Privacy Preserving Data Mining,” KDUbiq Summer School, 2006.

Privacy-Preserving Data Mining (PPDM)



**Compare, match, and analyze data from different organizations
without disclosing the private data
to any other party**

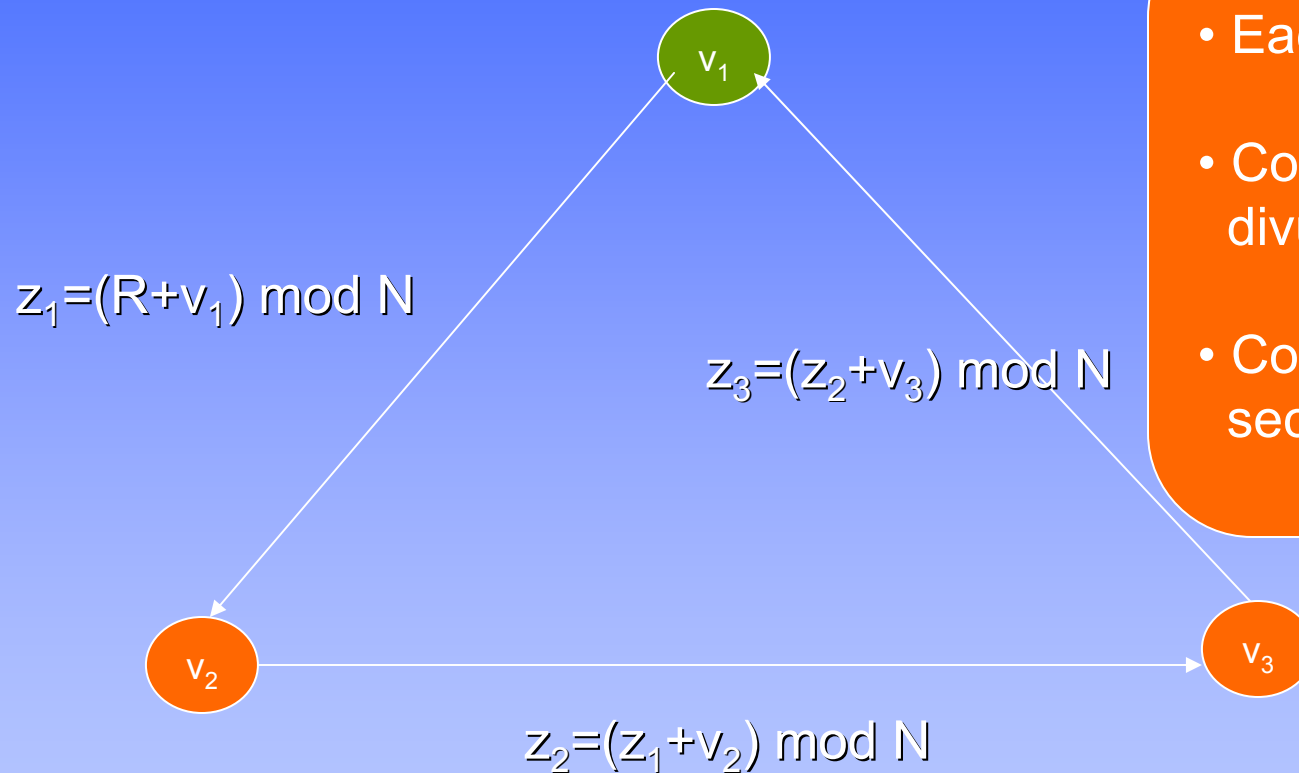
Blending Privacy-Preserving Techniques

- Some of the Existing Frameworks:
 - Data sanitization
 - Random additive noise (Agrawal and Srikant, 2001; Kargupta, 2003)
 - Random multiplicative noise (Liu, Kargupta, 2004)
 - Secured Multi-Party Computation (Goldreich, 1998, Clifton et al., 2003)
 - K-Anonymity (Sweeney, 2002)
 - Problems: Makes many assumptions about the user behavior.
 - Performs computations and communications as expected
 - Semi-honest
-

Game Theoretic Perspective

- Multi-Party PPDM as games
 - Related Work:
 - Halpern and Teague, 2004 explored Shamir's secret sharing problem from game theoretic perspective
 - Zhang et al, 2005
 - Abraham et al., 2006 extended their earlier work introduced generalized notion of nash equilibrium
 - Kargupta, Das, Liu, 2006
-

3-Party Secure Sum Computation



- Each party has a number
- Compute the sum without divulging the numbers
- Consider a sequence of secure sum operations.

R is uniformly distributed in $[0, N-1]$

Gaming Strategies

- Computing
 - Perform or do not perform the local computation.
 - Communication
 - Communicate or do not communicate with the necessary parties.
 - Privacy attacks on the data by individual users
 - Attack the messages received from other parties for divulging privacy-sensitive information regarding other parties.
 - Privacy attacks through collusion
 - Collude with others
-

Computing and Communication

- Let z_i be the number of computation node i performs out of m_i operations that an ideal party should perform.
 - If $z_i < m_i$ then it contributes
 - Error in overall estimation of the sum
 - Savings in computing load.
 - The effect of $(m_i - z_i)$ on the utility function can be assumed to be linear
 - Similar effect of the communication strategy
-

Individual Privacy Attacks

- ★ Define random variables W and Z as follows:

$$W = V + R, \quad Z = W \bmod N,$$

- ★ Can show that the joint probability mass function of Z, V ,
 $f_{ZV}(z, v) = f_Z(z)f_V(v)$.
- ★ Z and V are statically independent, and hence, Z does not contain any information about V .

Analysis of Collusion for Secure Sum

We can arrange the sites in the following order:

$$\underbrace{v_1 v_2 \dots v_{s-k-1}}_{\text{honest sites}} v_i \underbrace{v_{i+1} \dots v_{i+k}}_{\text{colluding sites}}$$

We have

$$\underbrace{\sum_{j=1}^{s-k-1} v_j}_{\text{denoted by X}} + v_i = v - \underbrace{\sum_{j=i+1}^{i+k} v_j}_{\text{denoted by C}},$$

where v is the total sum of the s values.

Continued

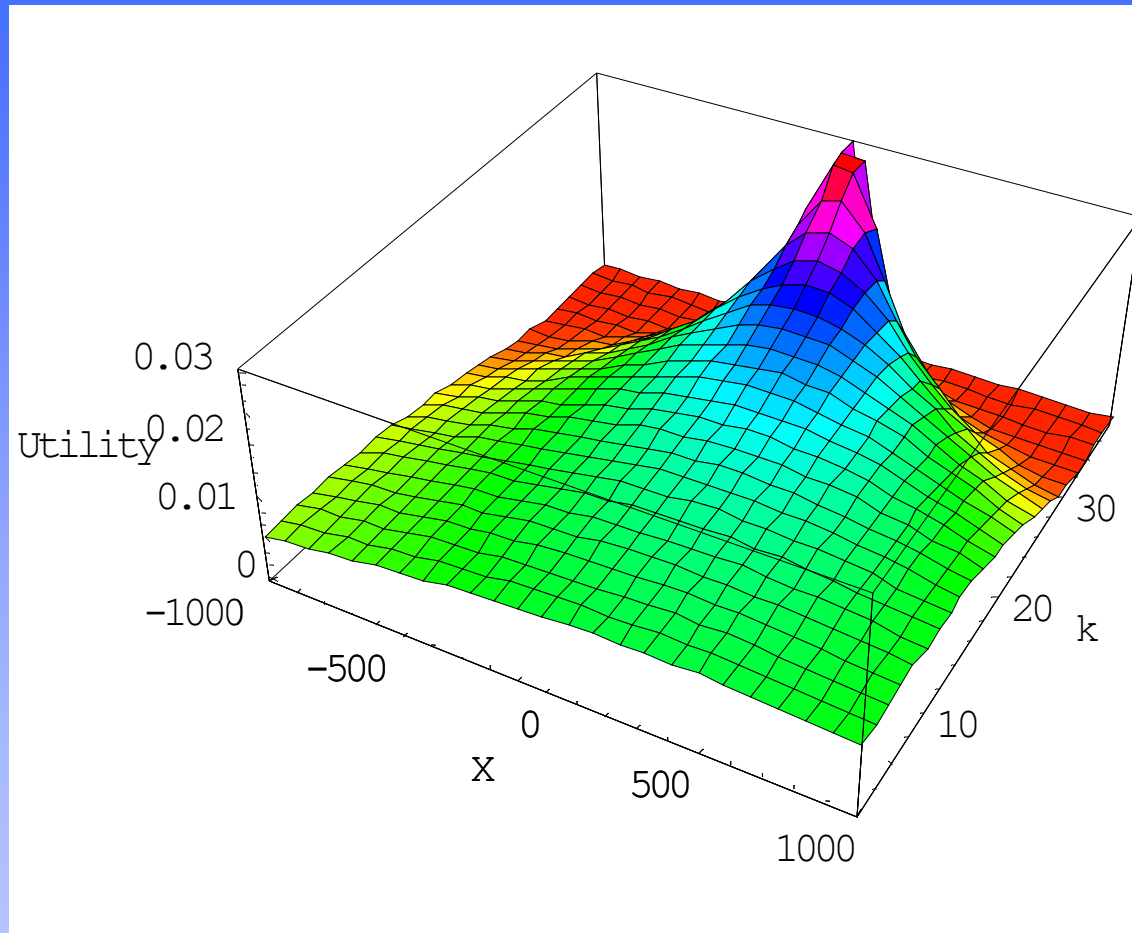
- ★ Each v_j ($j = 1, \dots, s$) is a discrete random variable independent and uniformly taking non-negative integer values over the interval $\{0, 1, \dots, m\}$.
- ★ Hence, X is the sum of $(s - k - 1)$ independent and uniformly distributed discrete random variables.



$$\textit{Privacybreach} = \textit{Posterior} - \textit{Prior}$$

$$= \frac{1}{(m+1)^{(s-k-1)}} \sum_{j=0}^r (-1)^j C_{(s-k-1)}^j C_{(s-k-1)+(r-j)(m+1)+t}^{(r-j)(m+1)+t} - \frac{1}{m+1}.$$

Utility Function Landscape



Overall Utility Function = f (Communication Cost, Computation Cost, Utility of Individual Privacy Attacks, Utility of Collusion)

Nash Equilibrium

- Nash equilibrium of a game guarantees that that if one player deviates from her equilibrium point, she cannot earn anything more if everybody else adhere to their equilibrium strategies.
 - What if more than one person deviate from the equilibrium strategy?
-

α -Resilient Equilibrium

- ★ An equilibrium is α -resilient if it tolerates deviations by coalitions of size up to α .
- ★ A joint action profile $(\sigma_1, \sigma_2, \dots, \sigma_n)$ is α -resilient if for any coalition of size less than or equal to α that deviates from the equilibrium, none of the members of this coalition do better than they do with action profile $(\sigma_1, \sigma_2, \dots, \sigma_n)$.

Analysis

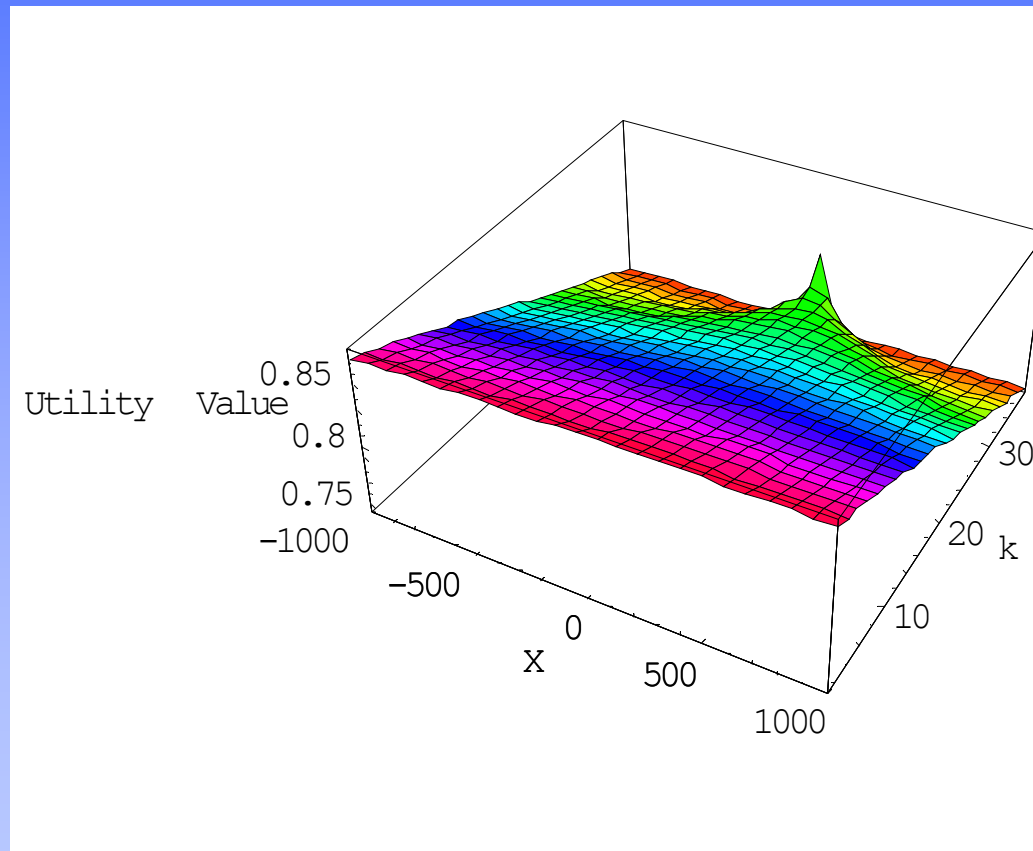
- The ideal scenario where each party abide by the rules of secure sum is not a equilibrium condition.
 - Each party will find the optimal number of colluding partners.
 - Any way out?
-

Cheap Talk and Punishment Strategies

- In game theory, Cheap talk (Farrell and Rabin, 1996) is pre-play communication which carries no cost but a threat to penalize if misbehavior is detected.
 - In order to form a colluding group i -th party needs to propose to j -th party. This may result in the following response:
 - j -th party accepts the invitation to collude
 - j -th party rejects the invitation to collude and invokes a punishment strategy
 - Example of a punishment strategy:
 - Split the local data in different random parts and perform a separate secure sum for each of these parts.
 - Increases the computation cost.
-

Cheap Talk in Action

- Change the location of the optima such that it corresponds to small value of k , the size of the colluding parties (ideally to $k=1$)



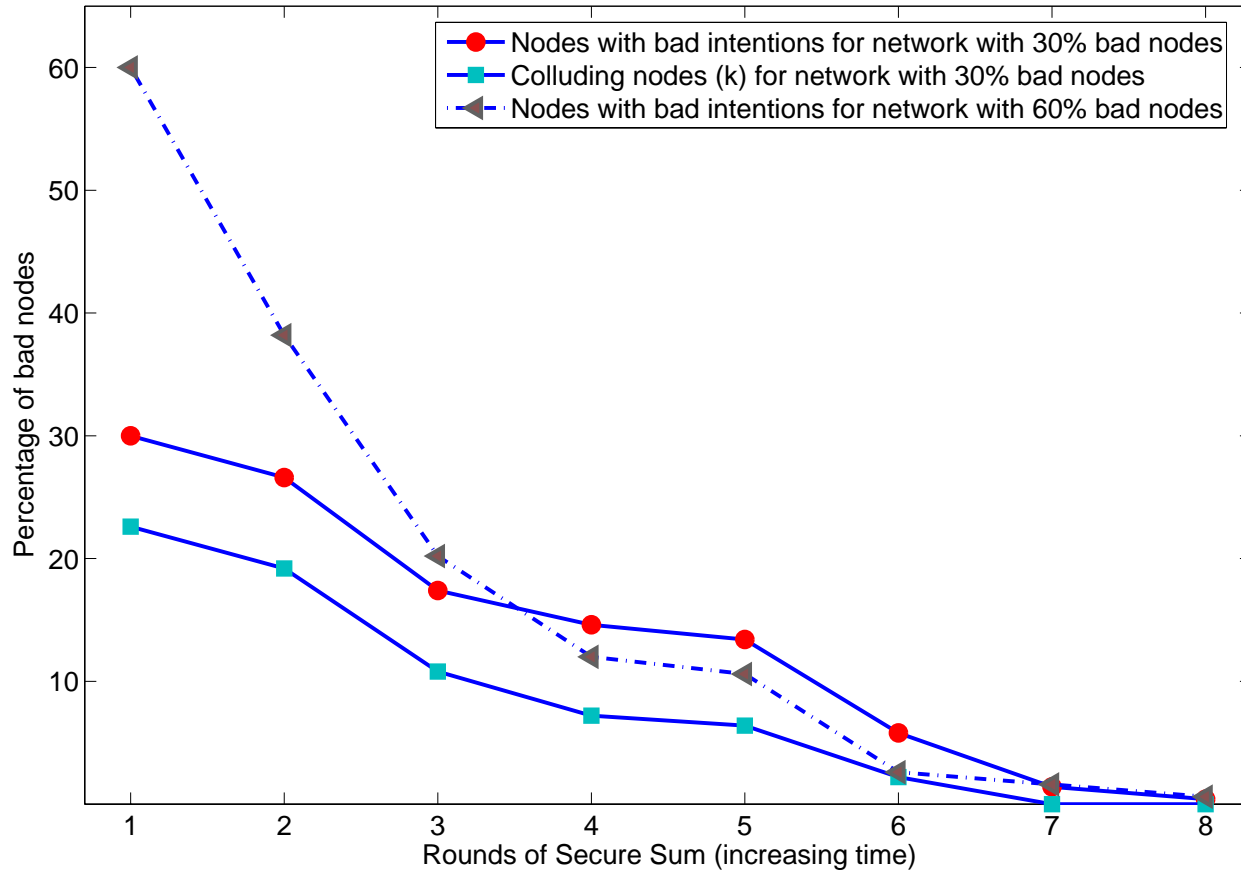
How to Implement Cheap Talk and Penalty Scheme?

- Centralized Approach
 - Decentralized Approach
 - Good nodes talk to their neighbors and warns about the penalty policy
 - Bad nodes send out invitations to collude
 - Penalty Scheme
 - Policy I: Remove the party from the multi-party privacy-preserving data mining application environment because of policy violation.
 - Policy II: if a party suspects a colluding group of size k' (an estimate of k) then it may split the every number used in a secure sum among k' different parts and demand k rounds of secure sum computation one for each of these k' parts.
-

Policy II: Randomly Split Shares

- Each good node (i) has its local estimate of the maximal size (k_i) of the colluding party.
 - Node i splits its local value v_i among k_i random shares and run a separate secure sum algorithm for each share.
 - Penalty mechanism works in a decentralized asynchronous manner.
-

Simulation Results



- Simulation with 500 nodes; Converges to equilibrium state corresponding to no collusion.

Conclusions

- P2P Data Mining: An emerging area of distributed data mining with many potential applications
 - Local algorithms
 - Exact local algorithms
 - Approximate local algorithms:
 - Probabilistic approximation
 - Deterministic approximation
 - Privacy is an important issue in P2P data mining
 - Privacy-preserving data mining algorithms need to meet the real-life challenges.
 - Game theory offers an interesting avenue
-