Quality-Aware Probing of Uncertain Data with Resource Constraints

Jinchuan Chen

The Hong Kong Polytechnic University
csjcchen@comp.polyu.edu.hk

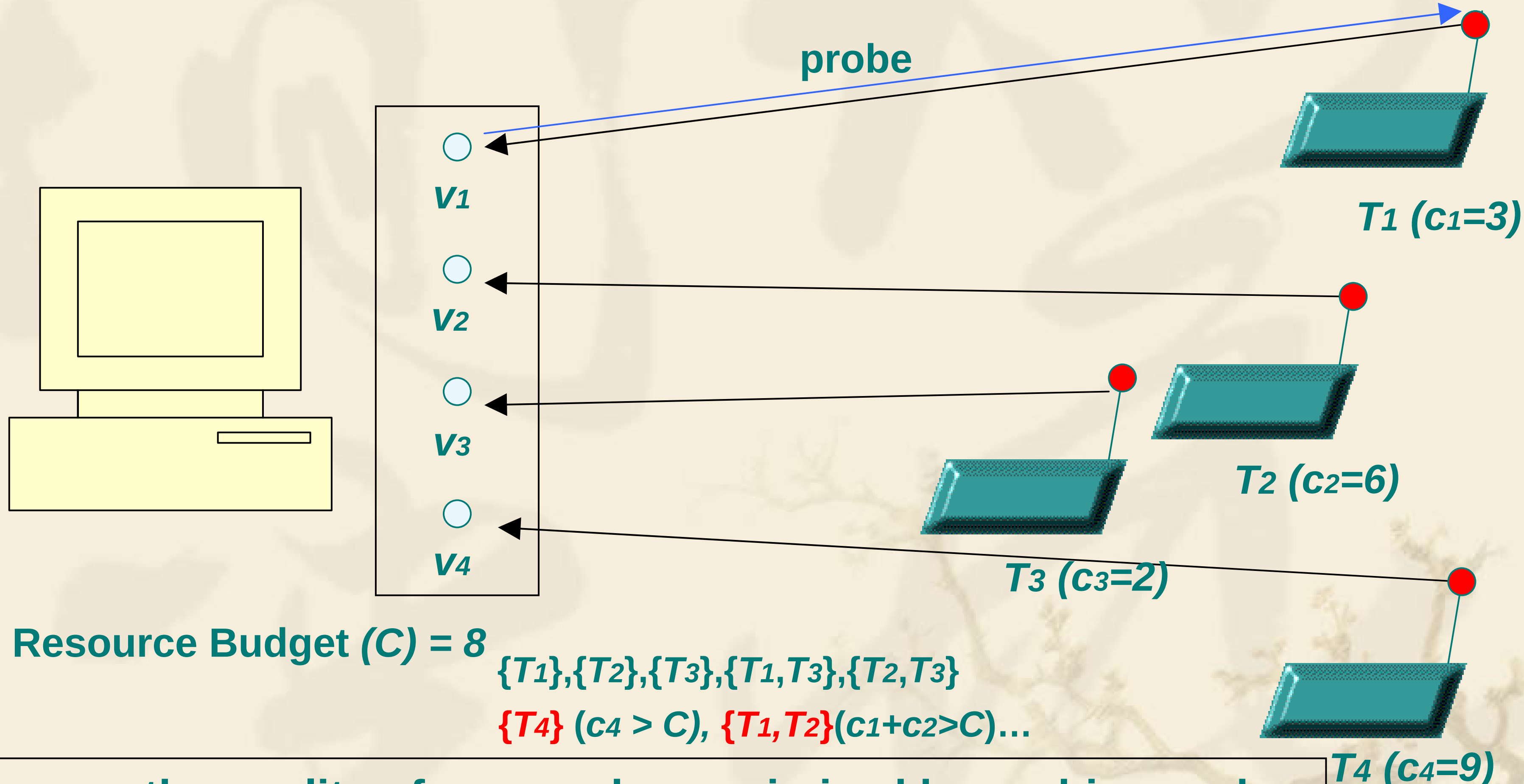
Reynold Cheng

The University of Hong Kong
ckcheng@cs.hku.hk

20th International Conference on Scientific and Statistical Database Management (SSDBM)
July 9-11, 2008, Hong Kong, China

Sensor-based Applications

Problem Overview



Resource Budget *(C) = 8*

{T_1},{T_2},{T_3},{T_1,T_3},{T_2,T_3}

{T_4} (c_4 > C), {T_1,T_2}(c_1+c_2>C)…

How can the quality of a query be maximized by probing under limited resource constraints?

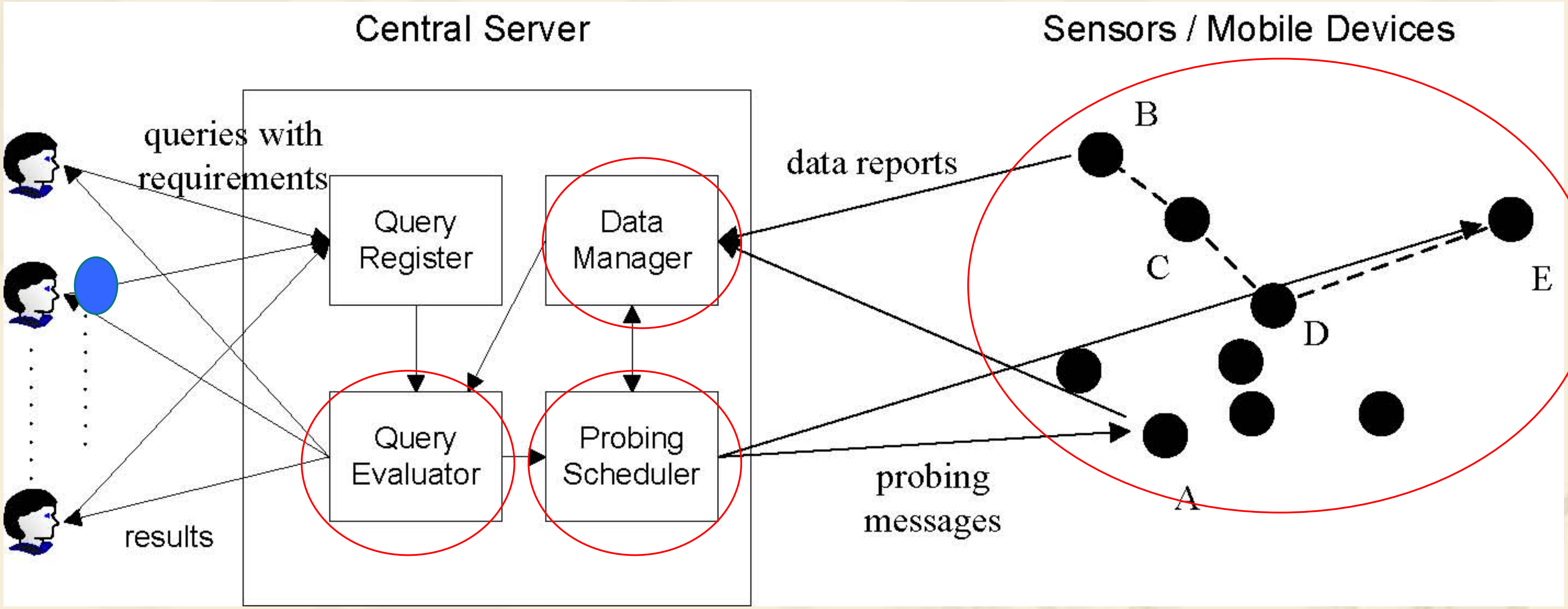
Related Works

- Probing Plans
  - [VLDB00, SAC05, VLDB04b, ICDE05b, ICDE06]
- Probabilistic Queries
  - [SIGMOD03, VLDB04, ICDE05, RTS06, IS06, ICDE07, ICDE07b, TDS07 , ICDE08, SSDBM08]
- Uncertain Database Cleaning
  - [VLDB08]

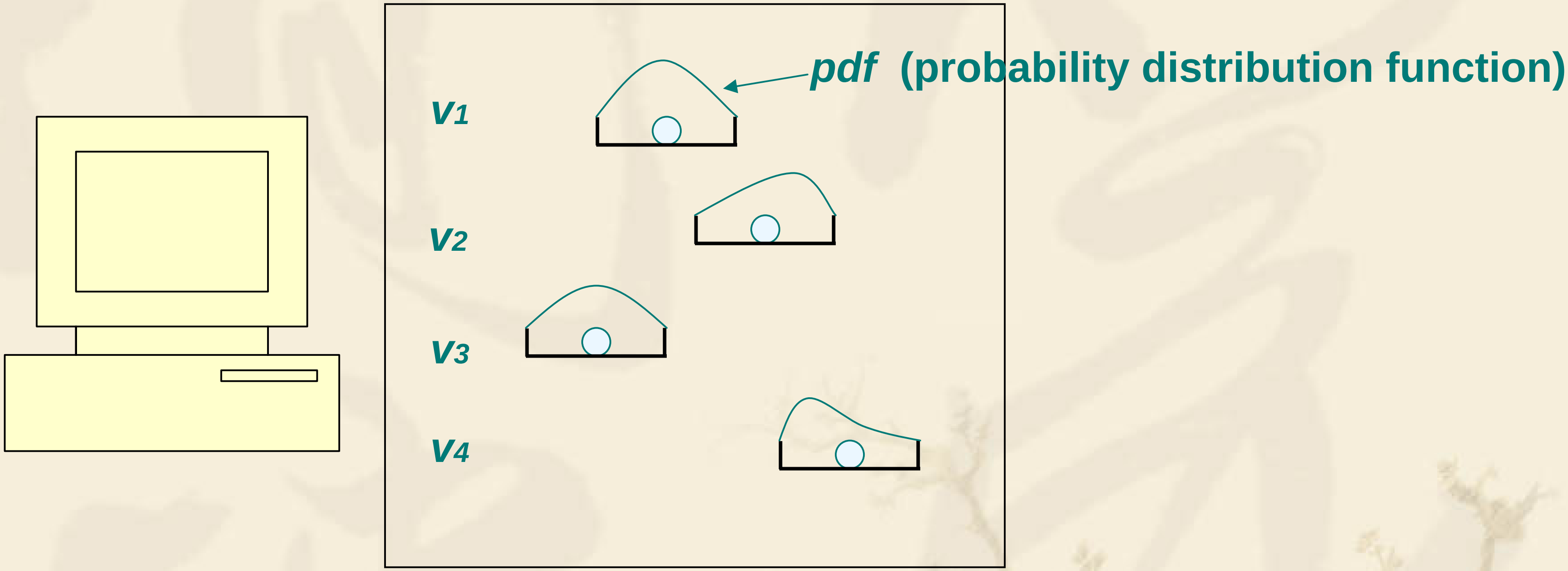
Talk Outline

- System Architecture
- Problem Formulation
  - Data, Query, Quality and Resource
- Quality-Aware Probing
  - The Single Query Problem
  - The Multiple Queries with Shared Budget Problem
- Experimental Results
- Conclusion

System Architecture



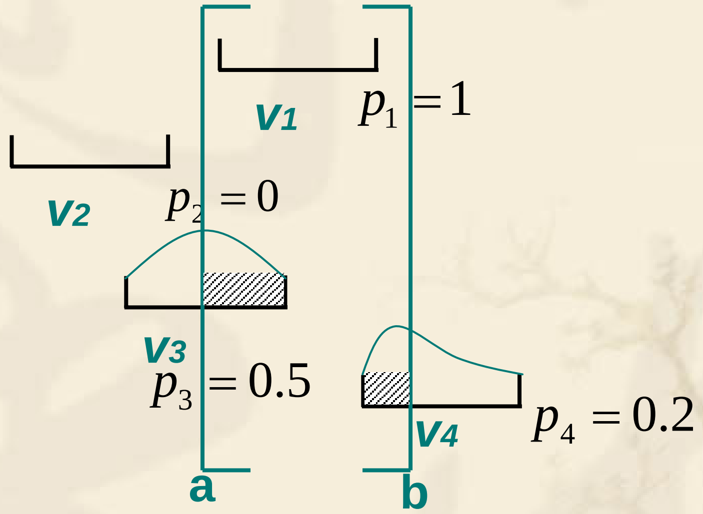
Uncertain Data Model



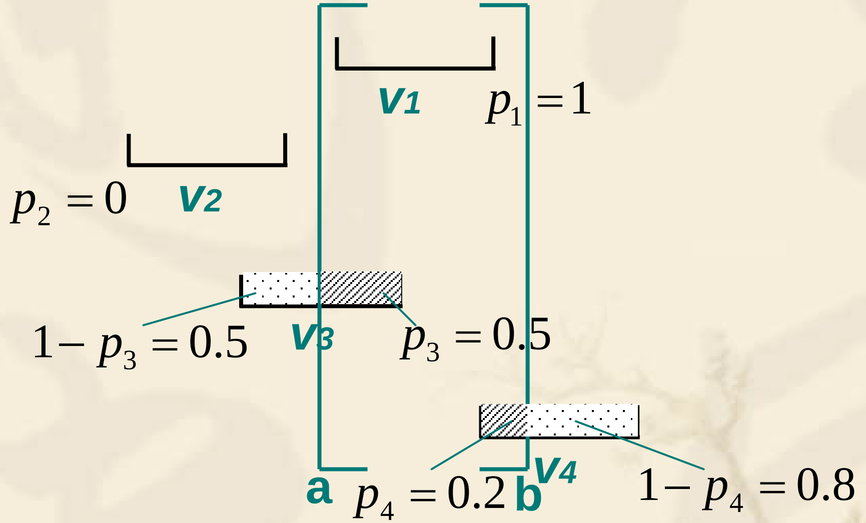
[VLDB04b, SSDBM99, IS06]

Probabilistic Queries

Definition: **P**robabilistic **R**ange **Q**uery (**PRQ**). Given a closed interval $[a,b]$, where $a,b \in R$ and $a \leq b$, a PRQ returns a set of tuples (T_i, p_i), where p_i is the non-zero probability that $T_i.v \in [a,b]$. **[SIGMOD03, VLDB04, ICDE07]**



Idea: Quality Score



"T_4 satisfies Q"

"T_4 does not satisfy Q"

Quality Score

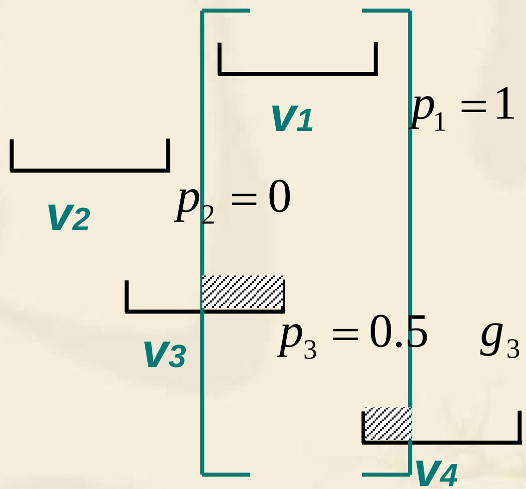
- The entropy of T_i for satisfying a PRQ is

$$g_i = -p_i \log p_i - (1 - p_i)\log(1 - p_i)$$

- Quality Score for a PRQ

$$H = -\sum_{i=1}^{n} (p_i \log p_i + (1 - p_i)\log(1 - p_i)) = \sum_{i=1}^{n} g_i$$

Quality Score (Example)



$p_1 = 1 \qquad g_1 = 1\log 1 = 0$

$p_2 = 0$

$p_3 = 0.5 \qquad g_3 = -2*(0.5\log 0.5) = 1$

$H = 0.81 + 1 = 1.81$

$p_4 = 0.2$

$g_4 = -0.2\log 0.2 - 0.8\log 0.8 = 0.81$

- Larger *H* implies lower quality
- *H* equals to zero if the result is precise (p_i = 0 or p_i = 1)
- No need to probe objects that leads to precise results
- Only needs to consider objects that satisfy the query with $p_i \in (0,1)$

Expected Quality

- To decide the sets of sensor(s) to probe, we choose the set that results in the best *expected* quality
- The set of sensors being probed can have different possible values.
  - Q may then have different results: $r_1, r_2, \ldots$
  - with corresponding probabilities $p(r_1), p(r_2), \ldots$
  - each result has a quality score $H_1, H_2, \ldots$
- The expected quality of probing this set
  - $p(r_1)*H_1 + p(r_2)*H_2 + \ldots$

Resource Budget

- Important resources for wireless sensor networks
  - power consumption
  - network bandwidth
- *no. of transmitted messages*
  - as a way for measuring these costs
  - also the *probing cost* in this paper
- each query Q has a *resource budget* C
  - max. # of transmitted messages allowed for improving H
- each item, T_i has a cost c_i
  - *# of transmitted messages spent for probing* T_i

Problem Modeling

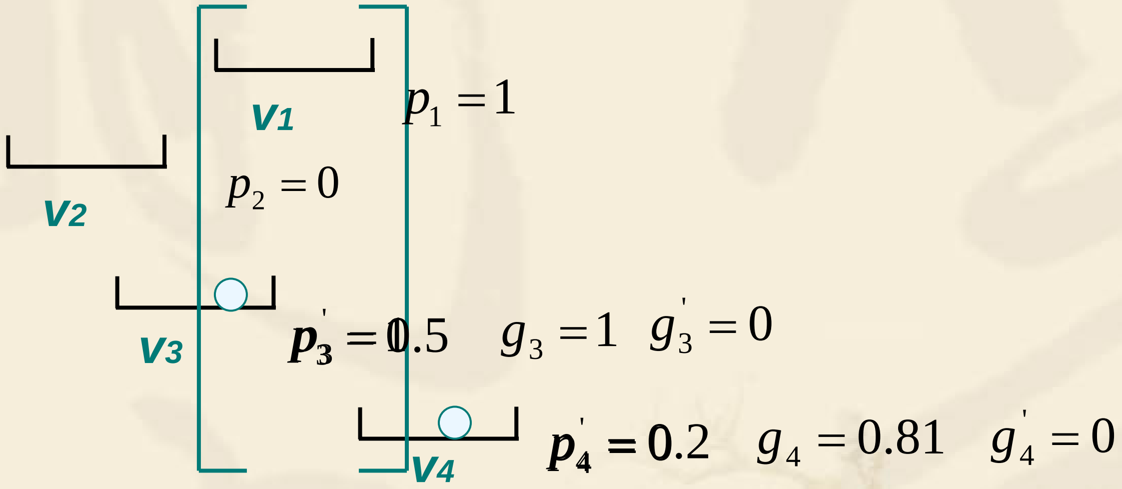
- Given
  - a query Q
  - a set of data objects $\{T_1,\ldots,T_n\}$ each of which is attached with a resource cost c_i
  - a method for calculating quality score H
  - a resource budget C,
- How to maximize the expected qu*ality*, i.e. obtain lowest H, with probing cost under C?

Brute-force Solution

- Brute-force solution
  - generate every subset of $\{T_1,\ldots,T_n\}$ whose costs are not larger than C
  - calculate the expected quality of probing this subset
  - select the one with the best expected quality

- Exponentially expensive in both computation and memory cost

Efficient Computation of Expected Quality Improvement



- The qualification probability for a probed data value is either 0 or 1.
- Probing reduces the uncertainty of objects to zero
- The expected quality improvement is exactly the **entropy** of the probed items.
  - e.g. expected quality improvement of probing $\{T_3, T_4\} = g_3 + g_4$

The Single Query Problem (SQ)

- Only **one** query *Q* is assumed when sensors being chosen.
- Based on our findings, we can formalize the problem as follows:

Maximize query quality **within budget *C***?

$$Maximize \sum_{i=1}^{n} x_i g_i$$

$$subject\ \ to \sum_{i=1}^{n} x_i c_i \le C$$

$$x_i \in \{0,1\}, i = 1,\ldots,n$$

- The expected quality improvement is exactly the entropy of the probed items

Dynamic-Programming Solution (DP)

- Denote the problem $P(C,N)$ and the optimal set $S = \{T_1, T_2, ..., T_m\}$
  - Consider sub-problem $P(C-c_1, N/\{T_1\})$
  - $S' = \{T_2, ..., T_m\}$ must be the optimal set for this sub-problem (proved in the paper)
  - leading to the *optimal substructure property*

Dynamic Programming Solution (DP)

Input An array of probing costs $c = (c_1,...,c_n)$

An array of gains $g = (g_1,...,g_n)$

The resource budget C

Output The optimal set

for $i := 1$ to n do

for $k := 1$ to C do

if $c_i > k$ or $v[k,i-1] > v[k-c_i, i-1] + g_i$

$v[k,i] := v[k,i-1]$

$s[k,i] := s[k,i-1]$

else

$v[k,i] := v[k-c_i, i-1] + g_i$

$s[k,i] := s[k-c_i, i-1]$

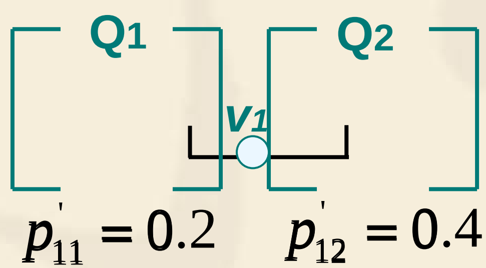
$s[k,i][i] := 1$

return $s[C,n]$

The Multiple Queries with Shared Budget Problem (MQSB)

- More than one query are processed at the server simultaneously
- A data item T_i may be involved in the results of multiple queries
- By probing T_i, all queries containing it in their results will have a better quality

Expected Quality Improvement of Probing T_i



- By probing T_1, both Q_1 and Q_2 will have a better quality
- Therefore, the expected quality improvement of probing T_i is the sum of its entropies for each query, i.e.

$$G_i = -\sum_{j=1}^{m} p_{ij} \log p_{ij} + (1 - p_{ij}) \log(1 - p_{ij})$$

Solution for MQSB

- The formal definition of MQSB has the same form as that of SQ.
- The only difference is the use of G_i to replace g_i.
- DP is also suitable for solving MQSB.

Approximate Solutions

- Greedy
  - Define efficiency as the amount of quality improvement obtained by consuming a unit of cost
  - Probe sensors in descending order of their efficiency until C is exhausted
- MaxVal
  - Probe sensors in descending order of their quality improvements until C is exhausted
- Random
  - Randomly choose an item to probe until C is exhausted

Computational Complexity

Algorithm	SQ	MQSB
DP	$O(nC)$	$O(nmC)$
Greedy	$O(n\log n)$	$O(nm\log nm)$
Random	$O(n)$	$O(nm)$
MaxVal	$O(n\log n)$	$O(nm\log nm)$

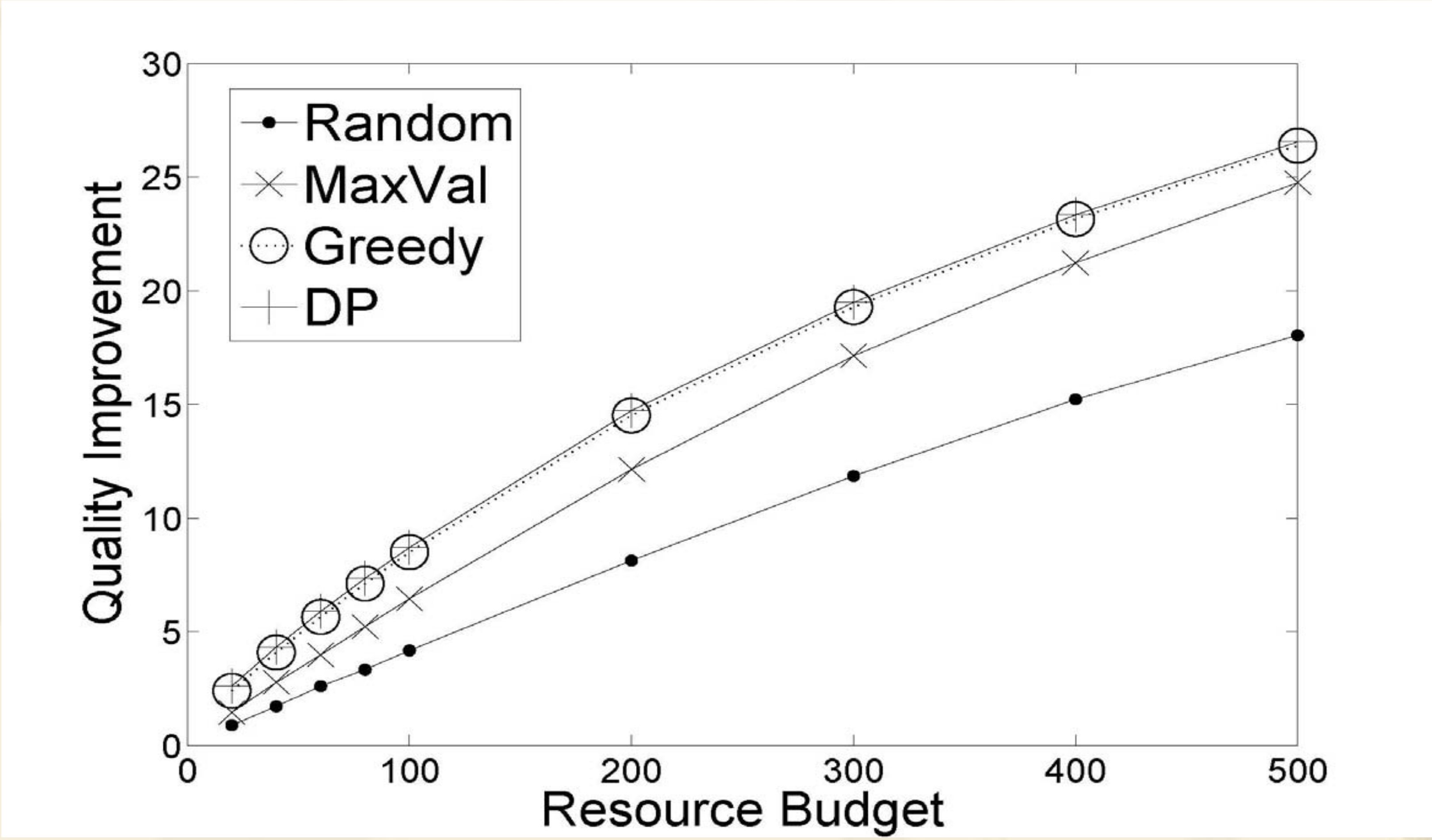
Memory Complexity

Algorithm	SQ	MQSB
DP	$O(n^2C)$	$O((nm)^2C)$
Greedy	$O(n)$	$O(nm)$
Random	$O(n)$	$O(nm)$
MaxVal	$O(n)$	$O(nm)$

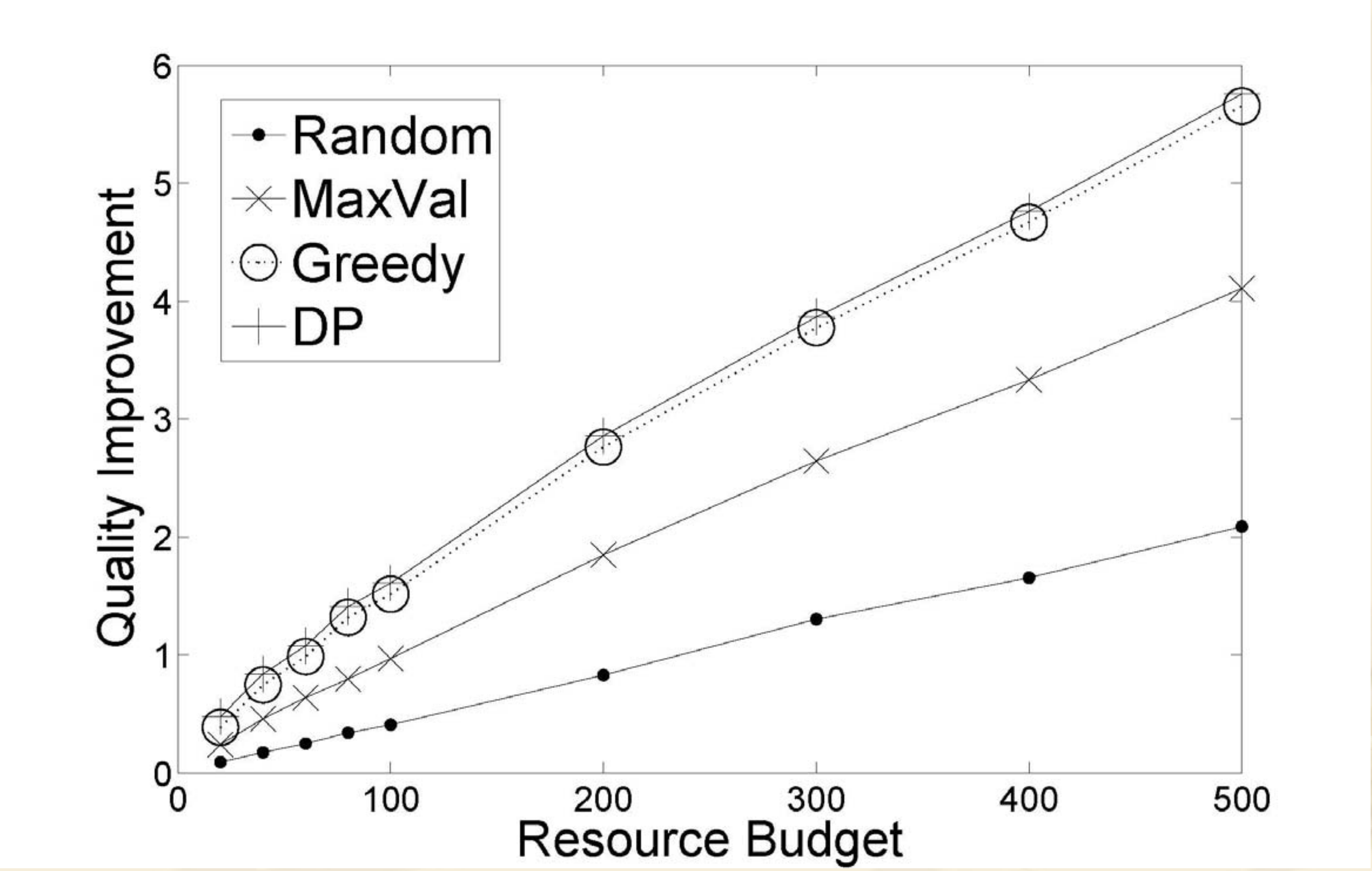
Experiment Setup

Uncertain Object DB	*Long Beach (53k)*
Uncertainty pdf	Uniform
Cost of Probing Sensors	Uniformly distributed in [1,10]
# of Queries (for MQSB)	10
Resource Budget	[20,500]

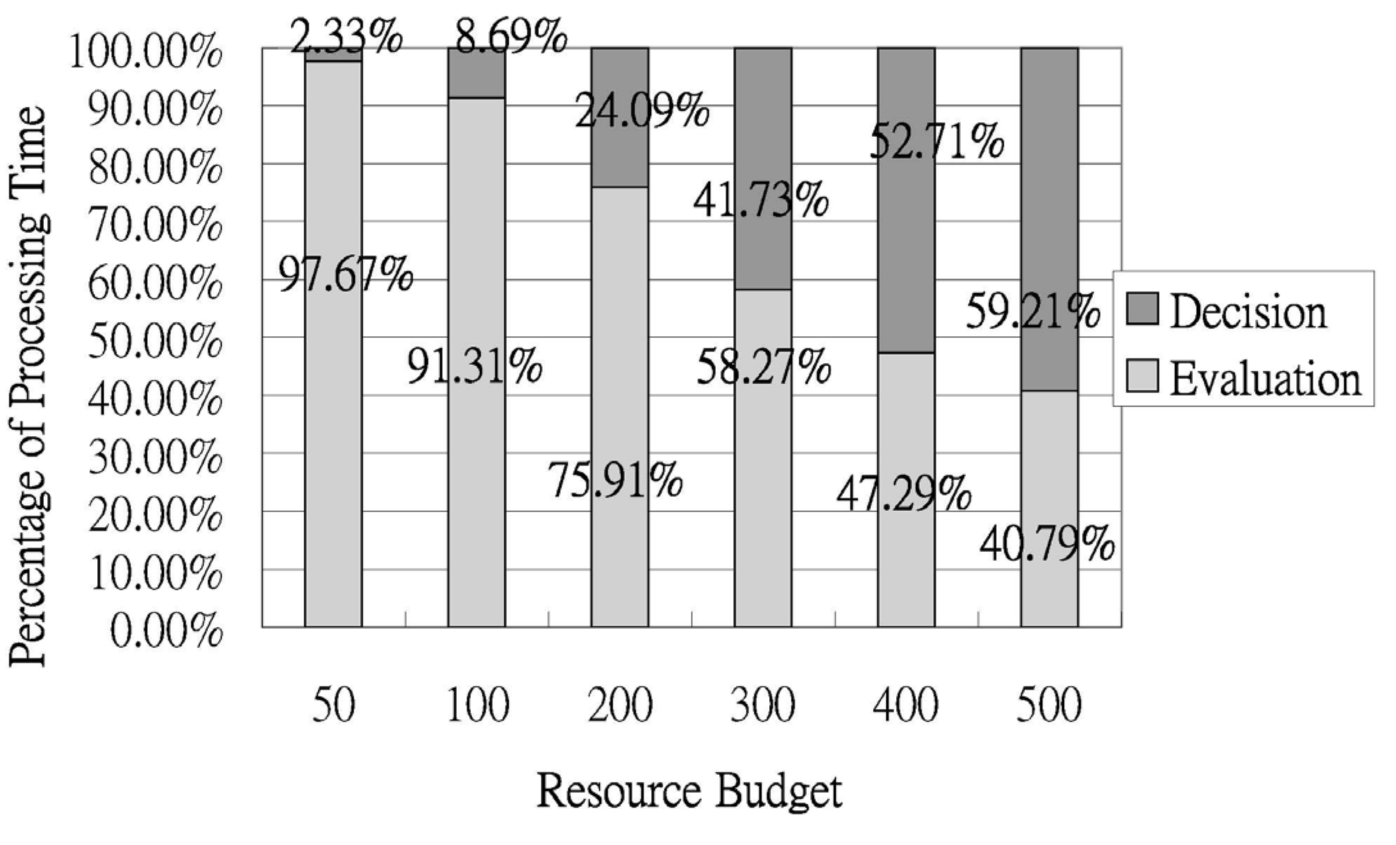
1. Quality Improvement vs. Resource Budget (SQ)



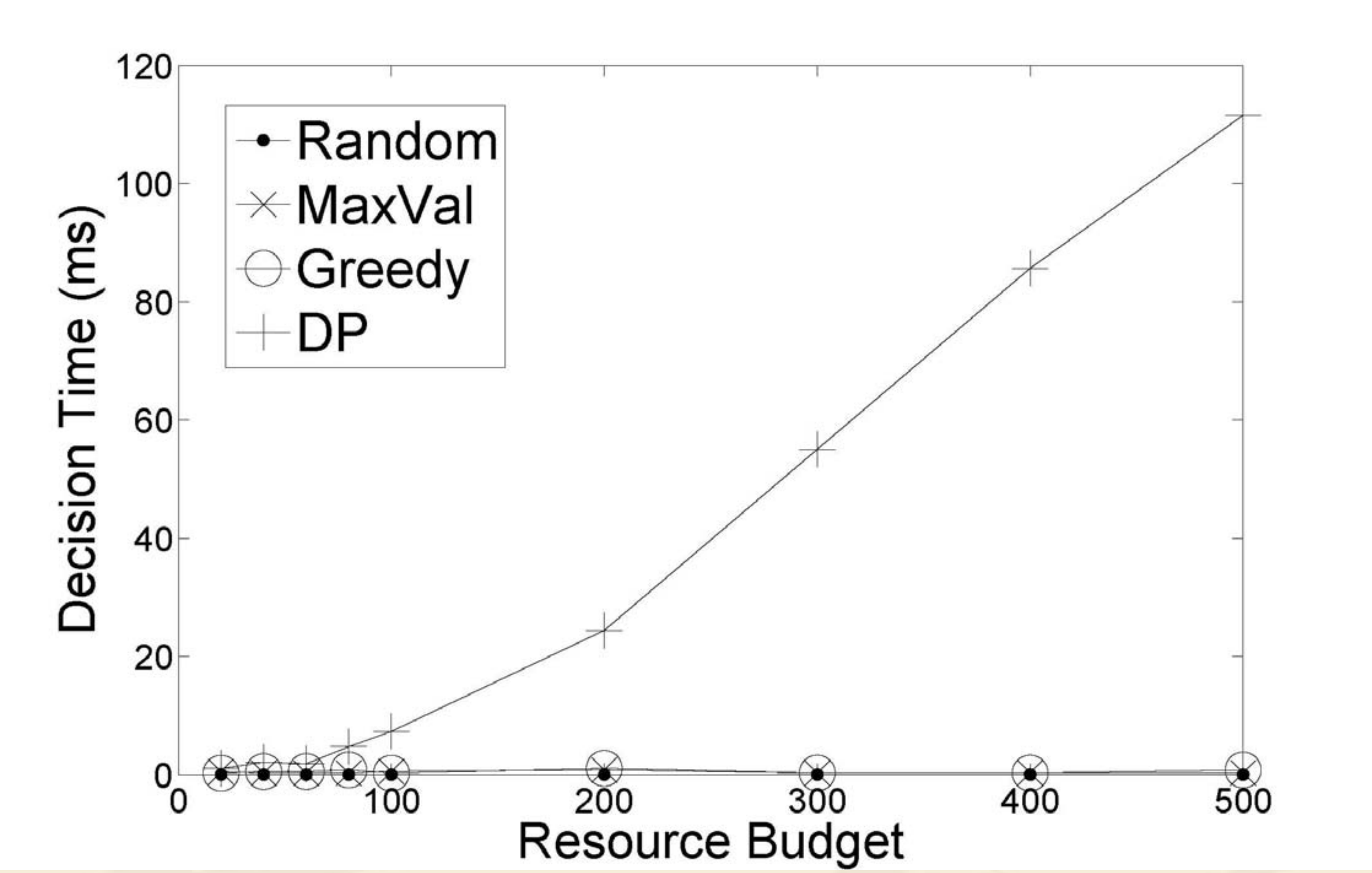
2. Quality Improvement vs. Resource Budget (MQSB)



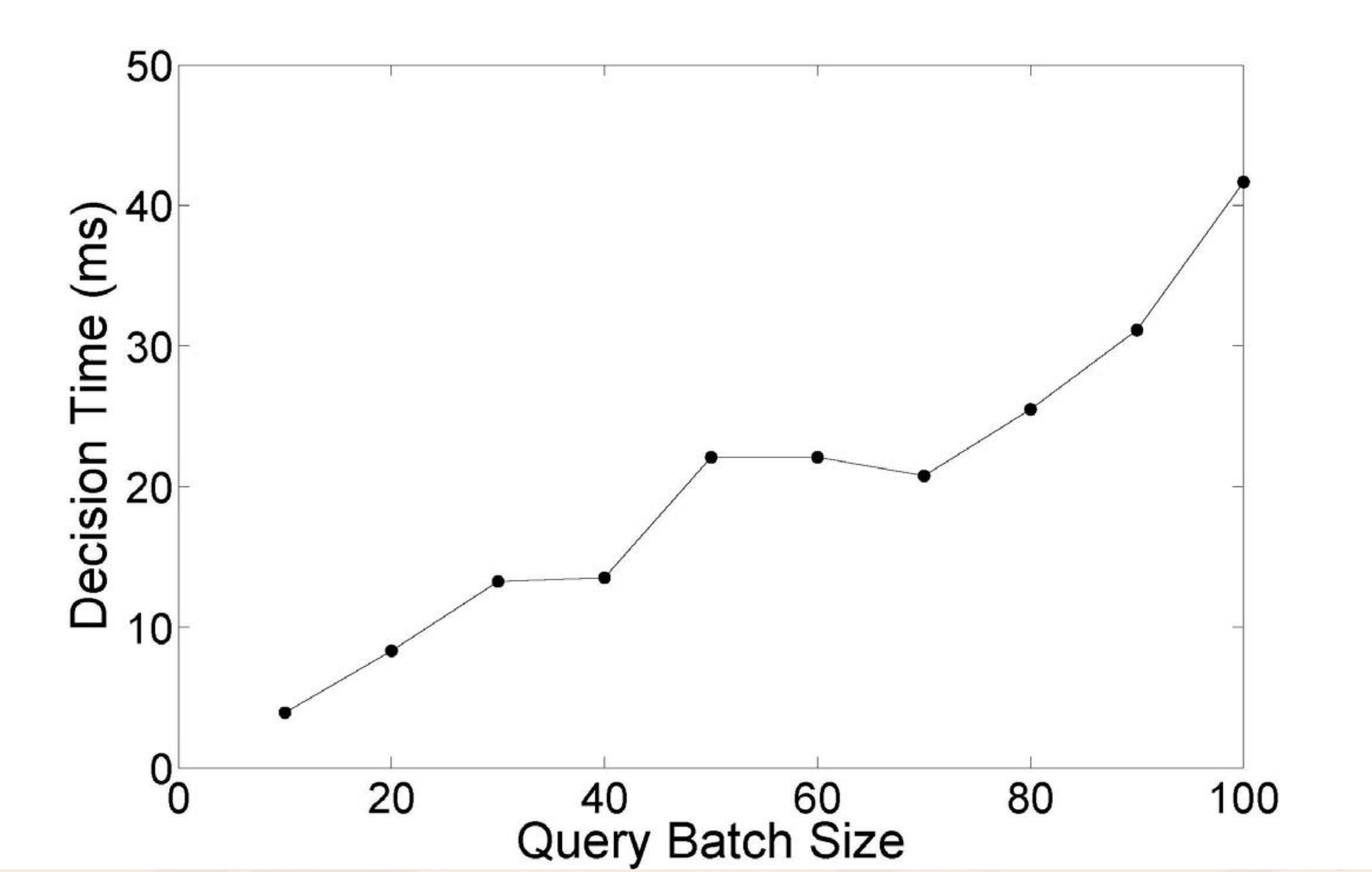
3. Time Analysis of DP



4. Decision Time vs. Resource Budget (SQ)



5. Scalability of Greedy for MQSB



References (1)

- **[ICDE07]** J. Chen and R. Cheng. Efficient evaluation of imprecise location-dependent queries. In ICDE, 2007
- **[ICDE08]** R. Cheng, J. Chen, M. Mokbel, and Chi-Yin Chow. Probabilistic verifiers: Evaluating constrained nearest-neighbor queries over uncertain data. In ICDE, 2008.
- **[SIGMOD03]** R. Cheng, D. Kalashnikov, and S. Prabhakar. Evaluating probabilistic queries over imprecise data. In Proc. ACM SIGMOD, 2003.
- **[ICDE05]** R. Cheng, Y. Xia, S. Prabhakar, and R. Shah. Change tolerant indexing on constantly evolving data. In ICDE, 2005.
- **[VLDB04]** R. Cheng, Y. Xia, S. Prabhakar, R. Shah, and J. S. Vitter. Efficient indexing methods for probabilistic threshold queries over uncertain data. In Proc. VLDB,2004.
- **[ICDE06]** David Chu, Amol Deshpande, Joseph Hellerstein, and Wei Hong. Approximate data collection in sensor networks using probabilistic models. In ICDE, 2006.
- **[VLDB04b]** A. Deshpande, C. Guestrin, S. Madden, J. Hellerstein, and W. Hong. Model-driven data acquisition in sensor networks. In VLDB, 2004.
- **[ICDE05b]** A. Despande, W. Hong C. Guestrin, and S. R. Madden. Exploiting correlated attributes in acquisitional query processing. In ICDE, 2005.
- **[SSDBM99]** D.Pfoser and C. Jensen. Capturing the uncertainty of moving-objects representations. In Proc. SSDBM, 1999.

References (2)

- **[ICDE04]** I. Lazaridis and S. Mehrotra. Approximate selection queries over imprecise data. In ICDE, 2004.
- **[ICDE07b]** V. Ljosa and A. Singh. APLA: Indexing arbitrary probability distributions. In ICDE, 2007.
- **[SIGMOD03b]** C. Olston, J. Jiang, and J. Widom. Adaptive filters for continuous queries over distributed data streams. In SIGMOD, 2003.
- **[VLDB00]** Chris Olston and Jennifer Widom. Offering a precision-performance tradeoff for aggregation queries over replicated data. In VLDB, 2000.
- **[TDS07]** Y. Tao, X. Xiao, and R. Cheng. Range search on multidimensional uncertain data. ACM Transactions on Database Systems, 32(15), 2007.
- **[IS06]** R. Cheng, D. Kalashnikov, and S. Prabhakar. Evaluation of probabilistic queriesover imprecise data in constantly-evolving environments. Information Systems Journal, 2006.
- **[MMOR03]** G. Diubin. The average behaviour of greedy algorithms for the knapsack problem: general distributions. Mathematical Methods of Operations Research, 57(3), 2003.
- **[SAC05]** Z. Liu, K. C. Sia, and J. Cho. Cost-efficient processing of min/max queries over distributed sensors with uncertainty. In SAC'05, 2005.
- **[VLDB08]** R. Cheng, J. Chen and X. Xie. Cleaning Uncertain Data with Quality Guarantees. To appear in Very Large Databases Conf. 2008
- **[SSDBM08]** Matthias Renz, Hans-Peter Kriegel and Thomas Bernecker. ProUD: Probabilistic Ranking in Uncertain Databases. In SSDBM 2008.

Conclusions

- We study the optimization issues of probabilistic query quality under limited budgets
- Solutions for both single and multiple queries are presented and experimentally evaluated
- Recently, we extend the study of the problem to a general probabilistic database model [VLDB08]
- We will investigate the problem for other queries

Thank you!

- **Contact:** Jinchuan Chen (csjcchen@comp.polyu.edu.hk)