Exploiting Spatial Autocorrelation to Efficiently Process Similarity Queries

Pusheng Zhang

Department of Computer Science and Engineering
University of Minnesota, Twin Cities
pusheng@cs.umn.edu

Advisers: Vipin Kumar and Shashi Shekhar

Thesis Committee: Sudipto Banerjee and Jaideep Srivastava

June 9, 2005
Biography Sketch

* Education
  - Ph.D. Student, Computer Science, U. of Minnesota, Minneapolis, MN, 2000 - Present
  - B.S., Computer Science, U. of Science and Technology of China, Hefei, China, 1999

* Honors and Awards
  - Student Paper Award, University Consortium of Geographic Information Science, 2005
  - Summer Fellowship, United Technologies Corporation, 2004
  - Student Paper Award, University Consortium of Geographic Information Science, 2003
  - Excellence in Research Recognition Award, U. of Minnesota, 2003
  - Excellent Student Fellowship, Government of Anhui Province, China, 1999
  - Excellent Student Scholarship, U. of Science and Technology of China, 1995-1998
Selected Publications

* Spatial and Temporal Databases

  - [NG2I05] Spatial Cone Tree: An Auxiliary Search Structure for Correlation-based Similarity Queries on Spatial Time Series Data, with S. Shekhar, Y. Huang, and V. Kumar, as a book chapter in an edited book, forthcoming
  
  - [TKDE05] Cone Tree: An Index Structure for Correlation Queries on Spatial Time Series Data, with S. Shekhar, V. Kumar, and Y. Huang, to be submitted to IEEE Transactions on Knowledge and Data Engineering
  
  - [SSTD03] Exploiting Spatial Autocorrelation to Efficiently Processing Correlation-Based Similarity Queries, with Y. Huang, S. Shekhar and V. Kumar, Symp. of Spatial and Temporal Databases, 2003
  
  - [PAKDD03] Correlation Analysis of Spatial Time Series Datasets: A Filter-and-Refine Approach, with Y. Huang, S. Shekhar and V. Kumar, Pacific-Asia Conference on Knowledge Discovery and Data Mining, 2003

* Spatial Data Mining

  
  - [KDD01] Graph-based Outlier Detection: Algorithms and Applications, with S. Shekhar, and C.T. Lu, ACM SIGKDD Int’l Conf. on Knowledge Discovery and Data Mining, 2001
Selected Publications

* Spatial Data Mining
  
  - [PAKDD05] Mining Time-profiled Associations, with J. Yoo and S. Shekhar, PAKDD 2005
  
  - [IDA03] Detecting Graph-based Spatial Outliers, with S. Shekhar, and C.T. Lu, J. Int. Data Analysis, 6(5), 2003
  
  
  - [ESDM05] Discovery of Patterns in Earth Science Data Using Data Mining, with M. Steinbach, V. Kumar, S. Shekhar, P. Tan, S. Klooster, and C. Potter, as a book chapter in Next Generation of Data Mining Applications
  
  - [ICTAI02] Data Mining for Selective Visualization of Large Spatial Datasets, with S. Shekhar, C.T. Lu, and R. Liu, ICTAI 2002
Spatio-Temporal Data are Everywhere

* Traditional Data
  * Numerical, categorical, ordinal, boolean, etc
  * e.g., city name, city population

* Spatio-Temporal Data
  * Spatial Attribute: geographically referenced
    – Location: e.g., longitude, latitude, and elevation
  * Temporal Information: time stamps

* Large spatial and temporal data are available
  * Advanced data collecting tools: GPS, satellites, sensors, retailers, etc
  * Information “nuggets”
    – Finding potentially useful information: business and scientific applications
    – e.g., Walmart transaction data, NASA climate data
Spatio-temporal Data Mining (STDM)

- The process of discovering
  - Interesting, potentially useful, non-trivial patterns
  - From large spatio-temporal datasets

- Application Domains
  - Eg: Climatology, Earth science, and Epidemiology

- Spatial Time Series (STS) Data
  - a spatial framework: set of locations
  - k attributes per location, each a time series of length m
Examples of Analysis Questions for STDM

⋆ Analysis Questions

⋆ Classification: predict long-term (3-6 months) rainfall
⋆ Clustering: find spatio-temporal homogeneous regions
⋆ Relationship: find locations influenced by El Niño (SSTD03, PAKDD03)
⋆ Anomaly: find unusual events (anomalies) (KDD01, GEOINFO03)

⋆ Relationship Mining:

⋆ Interest Measure
  – Correlation(time series \( f \), time series \( g \))
    – Range: \([-1, 1]\)
    – Uncorrelated if \( \rho = 0 \)


Correlation: 0.8831

©University of Minnesota
Example of Correlation Queries

* Correlation Queries:
  * STS Datasets: $D^1(location, f)$ and $D^2(location, g)$
  * Range Query: select $D^1.location$ from $D^1$ where $\text{corr}(D^1.f, \text{Query}) \geq \theta$

* All-Pair Query: finding all highly correlated pairs between two STS datasets
Example of Correlation Queries

* Correlation Queries:
  * STS Datasets: $D^1(location, f)$ and $D^2(location, g)$
  * Range Query: select $D^1.location$ from $D^1$ where $\text{corr}(D^1.f, \text{Query}) \geq \theta$
  * All-Pair Query: finding all highly correlated pairs between two STS datasets
Illustrative Application Domain

* Teleconnection: Ocean affects lands

* NASA Funded Project: *Discovery of Changes from the Global Carbon Cycle and Climate System Using Data Mining*

* For example: El Niño

  – Anomalous warming of tropical Eastern Pacific leads to:
    – Heavy Rain in Peru and Drought in Australia

  – D indicates drought;
  – R indicates unusually high rainfall
  – W indicates abnormally warm
Illustrative Application Domain

- El Niño: Anomalous warming of tropical Eastern Pacific
  - Characterized using time series
Illustrative Application Domain

* Teleconnection: Global Impact of El Niño

![Graph showing correlation between Nino 1+2 and land temperature]

Correlation: 0.84546

Correlation: 0.021829
Illustrative Application Domain

* Teleconnection: Global Impact of El Niño

![Graph showing correlation between teleconnection and land temperature](image)

* Computing Challenges
  * Large number of locations: e.g., 1 km by 1 km spatial resolution
  * Long time series, e.g., monthly data for 50 years – 600 observations
Big Picture

Data Mining Tasks:
- Classification
- Clustering
- Outlier Detection
- Relationship Mining

Correlation Queries

Query Language

DBMS
- Query Processing
- Storage and Indexing

Spatial
Time Series
Data
Highlights of Contributions

☆ How They Fit into the Big Picture

Data Mining Tasks:
- Classification
- Clustering
- Relationship Mining
- Outlier Detection

Correlation Queries

DBMS
- Query Language
- Query Processing
- Storage and Indexing

Spatial
Time Series
Data

Earth Science Data Mining
[SDM04] [ESDM 05] [PAKDD05]

Spatial Outlier Detection
[KDD01] [GEOINFO03] [IDA03]

Processing Correlation Queries
[SSTD03] [PAKDD03]

Indexing: Spatial Cone Tree
[NG2I05] [TKDE05]
Today’s Focus

* Spatial Cone Tree
  * A search structure on high-dim time series
  * Exploit spatial insights to design efficient query processing algorithms

* Speed up Correlation Query Processing
  * Range Queries
  * All-Pair Queries
Related Work

* Reduce Time Dimensionality
  * Transformations: e.g., DFT and DWT
  * Low-dim indexing: R-tree, Grid file, Quad-tree
  * e.g., F-index [Agrawal et al. 1993], [Rafiei et al. 2000], [Chan et al. 2003]

* Assumptions
  * Correlation is easy to computed in transformed space
  * Skewed power spectrum: few coefficients adequate
Limitations of Related Work

* Limitations
  * Effectiveness ↓ for non-skewed power spectrum
    - Non-skewed power spectrum: removing seasonality for time series
  
  ![Power Spectrum of DFT](image)

  Figure 1: Power Spectrum of DFT for: (a) Raw Time Series (b) Removing Seasonality

  * F-index is not efficient for (b)
  * Room to improve using spatial properties
Overview

- Motivation and Problem Definition
- Related Work and Contributions
- Proposed Approach
  - Evaluation of Proposed Approach
  - Conclusions & Future Work
Overview of Proposed Approach

- **Insight: Spatial Autocorrelation**
  - Nearby locations have similar time series

- **Spatial Cone Tree**
  - Group similar time series in space proximity together

- **Efficient Query Processing Algorithms**
  - Query processing based on cones (group of similar time series) instead of individual time series
  - Filter-and-refine paradigm
Spatial Autocorrelation

* Tobler’s first law of geography:
  * “Everything is related to everything else but nearby things are more related than distant things”

* Spatial autocorrelation
  * Nearby objects tend to be similar: possibly group similar ones together

Sea Surface Temperature (SST) in March, 1982
Normalization of Time Series to Unit Sphere

* Different Scales of Time Series – Normalization

* Time Series of length $m$, $f = \{f_1, f_2, \ldots, f_m\}$

* Normalized time series into unit vector $\hat{f}$

\[
\hat{f} = \frac{f - \bar{f}}{\sqrt{m\sigma_f}}
\]

* Fact 1: $\hat{f}$ is on surface of a $m$-dimensional unit sphere

\[
| f |^2 = \sum_{i=1}^{m} f_i^2 = \frac{\sum_{i=1}^{m} (f_i - \bar{f})^2}{m\sigma_f^2} = 1
\]

* Fact 2: $corr(f, g) = \cos(\hat{f}, \hat{g}) = \hat{f} \cdot \hat{g}$ for time series $f$, $g$

\[
corr(f, g) = \frac{\sum_{i=1}^{m} \left( \frac{f_i - \bar{f}}{\sqrt{m\sigma_f}} \right) \cdot \left( \frac{g_i - \bar{g}}{\sqrt{m\sigma_g}} \right)}{\sqrt{m\sigma_f^2} \sqrt{m\sigma_g^2}} = \hat{f} \cdot \hat{g} = \cos(\hat{f}, \hat{g})
\]
Normalization Illustration

* Time Series → Unit Vector in m-dim Sphere, e.g., m = 2

![Diagram showing unit vectors and their angles]

<table>
<thead>
<tr>
<th>Value(t₁)</th>
<th>Value(t₂)</th>
</tr>
</thead>
<tbody>
<tr>
<td>f</td>
<td>1</td>
</tr>
<tr>
<td>g</td>
<td>0</td>
</tr>
<tr>
<td>h</td>
<td>-1</td>
</tr>
<tr>
<td>k</td>
<td>1</td>
</tr>
</tbody>
</table>

* Correlation between Time Series → Angles between Unit Vectors
* Given: Time Series $f$, $g$ and correlation threshold $\theta$

* Correlation between Time Series $\rightarrow$ Angles between Unit Vectors
Spatial Autocorrelation Revisited

* Spatial autocorrelation
  * Nearby objects tend to be similar: possibly group similar ones together

* Insight: grouping similar objects based on space proximity

* Processing queries by groups instead of individuals

Sea Surface Temperature (SST) in March, 1982
Concepts of Cones

* Time Series → Unit Vector in m-dim Sphere

* Cone: Group of time series unit vectors
  * Axis unit vector: center vector
  * Span: largest angle between axis and any unit vector
Spatial Cone Tree

- A hierarchical search structure on normalized time series data
- Leaf node: cone and pointer to disk containing one or more normalized time series
- Internal node: cone and pointer to index page

![Tree Diagram]

M = 4  span_max = 30°
Operations on Spatial Cone Trees

* Query Operations:
  * Range query:
    – find highly correlated time series with query time series
  * All-Pair query:
    – find all pairs of highly correlated cross two spatial cone trees

* Maintenance Operations:
  * Insertion
  * Deletion
  * Bulk-loading
Illustration of Bulk-Loading

* Rationale and Strategy
  * Spatial autocorrelation
    - Nearby objects are related
  * Space-partition based bulk loading: e.g., quad-tree like
Range Query Processing

- Brute-force: Linear Scan

- Our Approach: Filter-and-Refine Strategy

Given:
- Min correlation threshold \( \theta \)
- Cone: axis and span
- Query \( q \)

Objective:
find all time series highly correlated with \( q \)
i.e., \( \text{corr}(t, q) > \theta \) or \( \text{angle}(t, q) < \arccos \theta \)

Key Idea:
Range Query Processing

- Brute-force: Linear Scan
- Our Approach: Filter-and-Refine Strategy

Given:
- Min correlation threshold $\theta$
- Cone: axis and span
- Query $q$

Objective:
find all time series highly correlated with $q$

i.e., $\text{corr}(t, q) > \theta$ or $\text{angle}(t, q) < \arccos\theta$

Key Idea:
- Determine upper bound and lower bound of angles between $q$ and time series in the cone
Range Query Processing

* Brute-force: Linear Scan

* Our Approach: Filter-and-Refine Strategy

Given:
- Min correlation threshold $\theta$
- Cone: axis and span
- Query $q$

Objective:
find all time series highly correlated with $q$
i.e., $\text{corr}(t, q) > \theta$ or $\text{angle}(t, q) < \arccos \theta$

Key Idea:
* Determine upper bound and lower bound of angles between $q$ and time series within the cone

* Upper Bound = $\sigma + \text{span}$; Lower Bound = $\sigma - \text{span}$. 
  $\sigma$ is the angle between the $q$ and the axis of the cone
∗ Eg: Given minimum correlation threshold \( \theta = 0.707 \), \( \arccos(\theta) = 45^\circ \)
Query Time Series:

Minimal Correlation Threshold = 0.8
An Example of Range Query Processing

Query Time Series:

Minimal Correlation Threshold = 0.8
An Example of Range Query Processing

Query Time Series:

Minimal Correlation Threshold = 0.8
An Example of Range Query Processing

Query Time Series:

Minimal Correlation Threshold = 0.8
An Example of Range Query Processing

Query Time Series:

Minimal Correlation Threshold = 0.8
An Example of Range Query Processing

Query Time Series:

Minimal Correlation Threshold = 0.8
An Example of Range Query Processing

Query Time Series:

Minimal Correlation Threshold = 0.8
Hints on All-Pair Queries: Cone vs. Cone

Upper Bound $< 45$ degree  
All–True

Lower Bound $> 45$ degree  
All–False

Lower Bound $< 45$ degree  
Upper Bound $> 45$ degree  
Some–True
Overview

✓ Motivation and Problem Definition
✓ Related Work and Contributions
✓ Proposed Approach
⇒ Evaluation of Proposed Approach
  * Analytical evaluation with cost models
  * Experimental evaluations using real Earth science data
★ Conclusions & Future Work
Experimental Design: Hypothesis

* Efficiency:
  * Spatial Cone Tree vs. Linear Scan vs. F-Index

* Query Processing
  * Range Query
    – Eg: Finding region where its SST is highly correlated with NPP at Minneapolis
    – Vary different minimum correlation threshold
  * All-Pair Query
    – Eg: Finding all highly correlated pairs between SST and NPP
Experimental Evaluation: Data

* Workload

* SST: Monthly Sea Surface Temperature of Pacific
* NPP: Monthly Net Primary Production of USA
* Temporal Span: 1982-1993 (12 × 12 = 144)
* Spatial Resolution: 0.5° × 0.5°, | SST | = 11556, | NPP | = 2901

<table>
<thead>
<tr>
<th>Longitude</th>
<th>Latitude</th>
<th>SST (82-93)</th>
</tr>
</thead>
<tbody>
<tr>
<td>120.5W</td>
<td>5.0N</td>
<td>2560,2567, ..., 2787</td>
</tr>
<tr>
<td>121.0W</td>
<td>5.0N</td>
<td>2567,2456, ..., 2789</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
</tr>
<tr>
<td>179.5W</td>
<td>5.0S</td>
<td>2034,2175, ..., 2445</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Longitude</th>
<th>Latitude</th>
<th>NPP (82-93)</th>
</tr>
</thead>
<tbody>
<tr>
<td>97.0W</td>
<td>33.5N</td>
<td>4.56,5.67, ..., 6.90</td>
</tr>
<tr>
<td>97.0W</td>
<td>34.0N</td>
<td>4.34,6.29, ..., 7.56</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
</tr>
<tr>
<td>97.0W</td>
<td>38.0N</td>
<td>2.34,3.23, ..., 4.34</td>
</tr>
</tbody>
</table>
Results of Range Query Processing vs. Linear Scan

* Variable Parameters
  * Minimal correlation threshold $\theta$: 0.3-0.9

![Graph showing correlation cost vs. minimum correlation threshold]

- Computational cost is reduced to 55% – 11%
- Speedup of factor 2 – 10
- Increase minimum correlation threshold $\Rightarrow$ computational savings $\uparrow$
Results of All-Pair Queries vs. Linear Scan

★ Variable Parameters

★ Minimal correlation threshold $\theta$: 0.3, 0.5, 0.7, 0.9

★ Computational cost is reduced to 63% – 2 %

★ speedup of factor 1.6 – 50

★ Increase minimum correlation threshold ⇒ computational savings ↑
Comparison with F-Index

* F-Index Revisited
  * DFT: time domain \(\Rightarrow\) frequency domain
  * Obtain the coefficients of discrete Fourier transforms of every time series
  * Build a \(2d\) R*-tree index using the first \(d\) coefficients (1-3)
  * Query processing strategy using R*-tree: minimum bounding boxes (MBRs)

* Filter-and-Refine Query Processing
  * Filter out false candidates based on intersections of MBRs
  * Approximation search: superset of the answers
  * Refinement: manual checking of superset
Analytical Cost Models – Notations

* Notations
  * $n$: number of time series
  * $m$: length of time series
  * $L$: number of nodes in the tree (typically $L \ll n$)
  * $\epsilon$: corresponding normalized Euclidean distance for a given $\theta$
  * $f_c$: number of DFT coefficients used in F-Index
  * $c$: average number of entries per node
  * $a_i$: average side size for node $i$
  * $S_{findex}$: percentage of time series in refinement for F-index
  * $S_{sct}$: percentage of time series in refinement for Spatial Cone Tree
  * $t_{mbr}$: average cost of MBR intersection test
  * $t_{corr}$: cost of a correlation between two time series
Cost Models for F-Index and Spatial Cone Trees

* Both Use Filter-and-Refine Strategy
  * F-Index uses R-tree intersection with query time series
  * Spatial cone tree uses All-True and All-False filtering

* F-Index:
  * Filtering: \( \sum_{i=1}^{L_{findex}} (a_i + \epsilon)^2 f_c = L_{findex} \times S_{findex} \times t_{mbr} \)
  * Refinement: \( n \times S_{findex} \times t_{corr} \)
  * selectivity \( S_{findex} \): related to query, R*-tree, and \( f_c \)

* Spatial Cone Tree
  * Filtering: \( L_{sct} \times S_{sct} \times t_{corr} \)
  * Refinement: \( n \times S_{sct} \times t_{corr} \)
  * selectivity \( S_{sct} \): related to query and Spatial Cone Tree
Spatial Cone Tree vs. F-Index

* Dominant Zone for Computation
  * Refinement step is often more expensive than Filtering Step
  * Selectivity ($S_{findex}$ or $S_{sct}$) is an important factor
    - data with skewed power spectrum or $f_c \uparrow$: $S_{findex} < S_{sct}$ (F-Index wins)
    - spatial data with non-skewed power spectrum: $S_{findex} > S_{sct}$ (SCT wins)

* Evaluation
  * Check the refinement cost for both methods using selectivity
    - Check the percentage of time series in the refinement step
Comparison with F-Index with Range Query

- F-Index: Select 2, 6, 8, 10 coefficients for R*-Tree respectively

- Spatial Cone Tree outperforms the F-Index for the spatial time series data with non-skewed power spectrum.
Recap of Evaluation

* Analytical Analysis using Cost Models

* Experimental Evaluation

  * Hypothesis: efficiency (Spatial Cone Tree vs. Linear Scan vs. F-Index)
  * Workload: NASA Earth science data
  * Comparison with Linear Scan
    - Speedup of factor of 2–10 for range queries
    - Speedup of factor of 1.6–50 for all-pair queries
  * Comparison with the F-Index
    * Outperforming F-Index for spatial time series data with non-skewed power spectrum
    * F-Index might be better for data with skewed power spectrum
  * Increase minimum correlation threshold $\theta \Rightarrow \text{speedup} \uparrow$
Conclusions

* Spatial time series data and correlation-based queries are abundant in many applications
  * e.g., NASA Earth science data, epidemiology, climatology

* Spatial Cone Tree
  * An indexing structure on spatial time series data
  * Support spatial time series data type and operations
  * Spatial autocorrelation facilitated query processing
  * Analytical evaluation and experimental evaluation
Other Contributions

* Spatial Data Mining: Spatial Outlier Detection
  * Outlier detection: e.g., credit card fraud detection
  * Spatial outliers: neighbor-based outlier detection in topological data
  * “A Unified Approach to Detecting Spatial Outliers”, GeoInformatica, 7(2), 2003

* Visualization of Spatio-Temporal Data
  * “Data Mining for Selective Visualization of Large Spatial Datasets”, IC-TAI 2002

* Data Mining with the Application of Intrusion Detection
  * Summer Research Intern at the United Technologies Research Center, 2004
Future Work: Short Term

Data Mining Tasks:
Classification  Relationship Mining
Clustering  Outlier Detection

DBMS
Query Language
Query Processing
Storage and Indexing

Spatial
Time Series
Data

Correlation Queries

Accomplished
Future Work (Short Term)
Long Term Agenda: Spatio-Temporal Databases and Data Mining

* Spatio-Temporal Database and Data Mining Design
  * Many applications, such as GIS, epidemiology, climatology, etc

* Challenges:
  * Spatial feature selections
  * Indoor space modeling
  * Navigation routing
  * Continuous query processing and optimization
  * Mining frequent common preferences
Acknowledgements

★ Advisers: Prof. Shashi Shekhar and Prof. Vipin Kumar
★ Committee: Prof. Jaideep Srivastava and Prof. Sudipto Banerjee
★ Spatial Databases and Data Mining Group Members
★ NASA Collaborators: Chris Potter and Steve Klooster
★ Dept. of Comp. Sci. & Eng. and AHPCRC at U. of M.
★ Doctoral Dissertation Fellowship by the Graduate School
Thank You! and Questions

URL: http://www.cs.umn.edu/~pusheng
Email: pusheng@cs.umn.edu
Correlation: Review

* Definition

\[ \rho = \frac{\sigma_{f,g}}{\sigma_f \sigma_g} = \frac{1}{m} \sum_{t=1}^{m} \left( \frac{(f_t - \bar{f})(g_t - \bar{g})}{\sigma_f \sigma_g} \right) \]

* Statistical Significance of Correlation

  * How large is significant for correlation?
  
  * Significance Test
    
    - Fisher’s Z test: \( Z = \frac{1}{2} \log \frac{1+\rho}{1-\rho} \)
    
    - confidence level \( \Rightarrow \rho_{\text{min}} \)
      
      - E.g.: confidence level 95% \( \Rightarrow \rho_{\text{min}} = 0.46 \)
      
    - Student-t test for short time series
Cost Model (backup)

* L: Number of nodes for R*-tree

\[ L = \frac{n}{c_{rtree}} + \frac{n}{c_{rtree}^2} + \ldots + \frac{n}{c_{rtree}^{h-1}} + 1 = \frac{n}{c_{rtree}^{h-1}} \times (1 - \frac{1}{c^{h-1}}) \]

* height of the R*-tree: \( h = 1 + \left\lceil \frac{N}{c_{rtree}} \right\rceil \)
Problem Definition

* Given:
  * Large spatial time series datasets
  * A set of operations
    – Correlation-based Queries:
      – Range query, all-pair queries, and nearest-neighbor query
      – Maintenance Operations: insertion, deletion, bulk-loading

* Find: A disk-based data structure

* Objectives: computational efficiency

* Constrains:
  * Correctness and Completeness
    – No false admissions
    – No false drops
Discussion on Cone Span

- Given minimum correlation threshold $\theta$: $(\theta, 1]$
  - Corresponding angle range is $[0 \arccos \theta)$

- The cone span should be less than $\frac{\arccos \theta}{2}$

- Empirical Estimation: Correlogram
Parameter Selection

* Correlogram
  * Distance vs. Correlation
  * Parameter Selection for Bulk Loading: Initial Cone Size

![Graph showing distance vs. correlation for parameter selection](image-url)
Design Issues

* Blocking Factor
  * the number of index records per disk page
  * Depend on length of time series (cone axis)
    – index record includes cone span, cone axis, and pointer
* Node Compression
  – Dimensionality reduction: reduce length of time series
  – Divide long time series into fixed-length smaller chunks

* Balancing Issue
  * Balanced tree: all leaf nodes are on the same level
  * Balancing is desirable
  * Overheads of keeping balancing are extensive
  * Explore balancing property
Completeness and Correctness

* Correctness

  * Lemma: Range query processing strategy is correct
    – Proof Sketch:
      – results are from all-true or refinement
      – all-true Lemma guarantees no false admission
      – refinement guarantees no false admission

* Completeness

  * Lemma: Range query processing strategy is complete
    – Proof Sketch:
      – dismissals happen in all-false or refinement
      – all-false lemma guarantees no false dismissals
      – refinement guarantees no false dismissals