Estruturas de Dados para Séries Temporais

Proposta de Dissertação

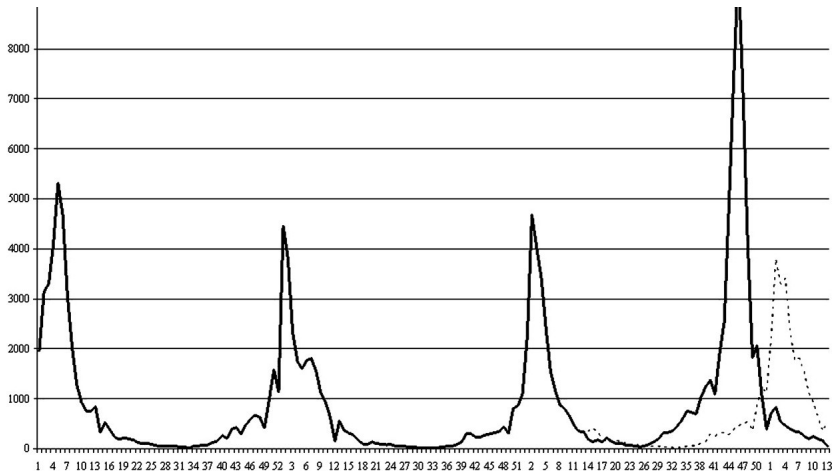
Caio Valentim
Orientador: Eduardo Laber
Co-Orientador: David Sotelo

Departamento de Informática - DI
Pontifícia Universidade Católica
Rio de Janeiro, Brasil

21 de Outubro, 2011

Motivação

- Séries temporais surgem em diversas aplicações com sísmica, finanças, meteorologia, dentre outras
- Identificar eventos pode ajudar a entender fenômenos que ocorrem ao longo do tempo na série temporal
- Séries temporais podem ser massivas



Definições

- Uma série temporal é uma sequência de n números reais $a_1, a_2, \ldots, a_n$
- Eventos são variações significativas em um dado intervalo de tempo

Formalização

Entrada: Uma série de números reais $a_1, a_2, \ldots, a_n$ (Offline)

Objetivo: Responder consultas, sobre a série de entrada, definidas por pares (t, d), onde t é um *inteiro* e d é *real* (Online)

Problema 1 - Todos os Pares
Encontrar todos os pares (i, j), $i < j$, tais que:

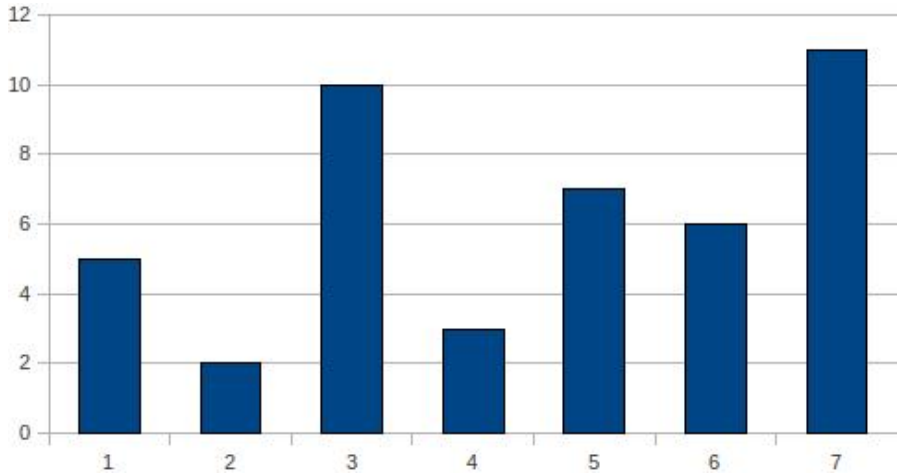
$$j - i \leq t \text{ e } a_j - a_i \geq d$$

Problema 2 - Inícios
Encontrar todos os i tais que existe ao menos um $j (i < j)$, tal que:

$$j - i \leq t \text{ e } a_j - a_i \geq d$$

Exemplo - Consulta Todos os Pares

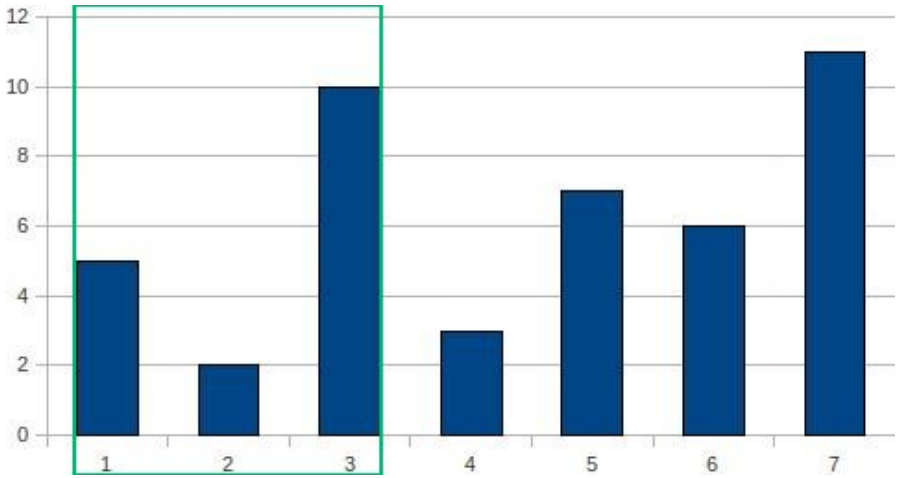
$$< 5, 2, 10, 3, 7, 6, 11 >$$



Consulta: $(t = 2, d = 5)$
Resposta: $\{\}$

Exemplo - Consulta Todos os Pares

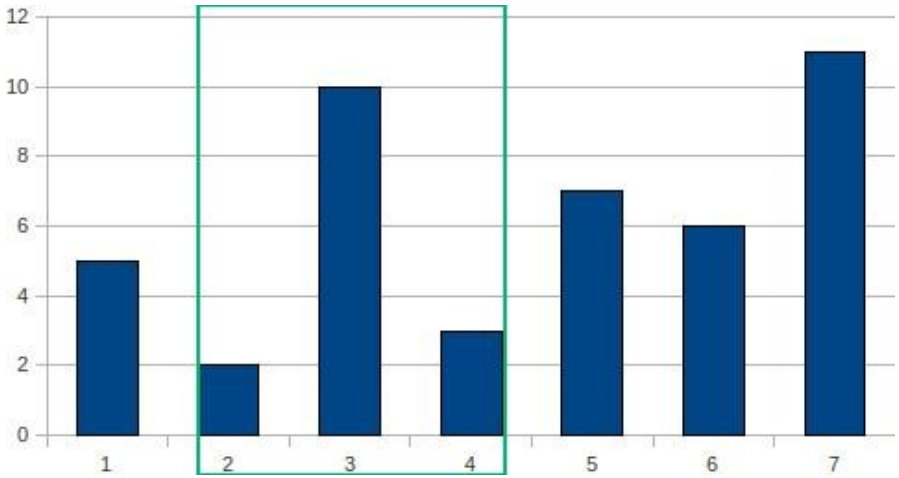
$$< 5, 2, 10, 3, 7, 6, 11 >$$



Consulta: $(t = 2, d = 5)$
Resposta: $\{(5, 10)\}$

Exemplo - Consulta Todos os Pares

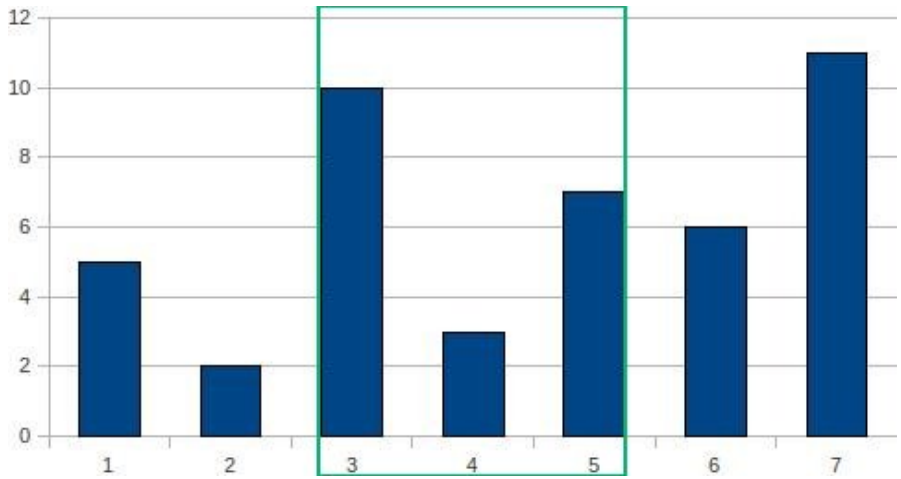
$$< 5, 2, 10, 3, 7, 6, 11 >$$



Consulta: $(t = 2, d = 5)$
Resposta: $\{(5, 10), (2, 10)\}$

Exemplo - Consulta Todos os Pares

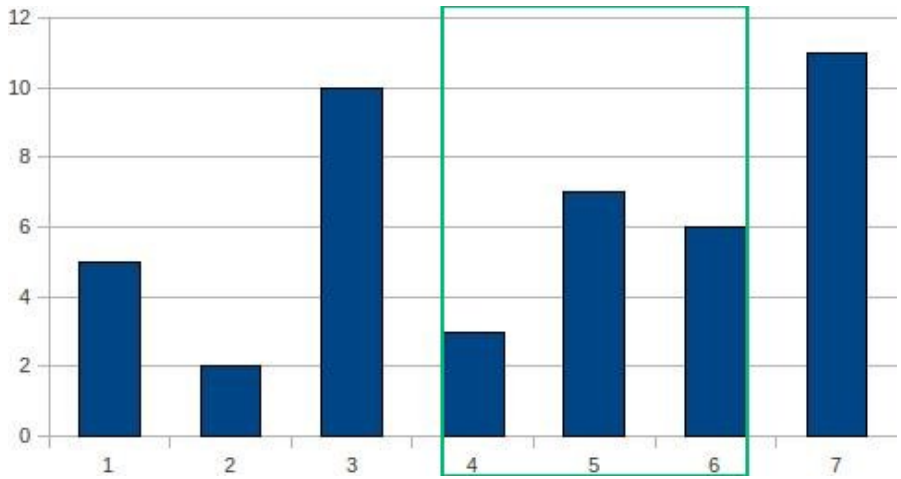
$$< 5, 2, 10, 3, 7, 6, 11 >$$



Consulta: $(t = 2, d = 5)$
Resposta: $\{(5, 10), (2, 10)\}$

Exemplo - Consulta Todos os Pares

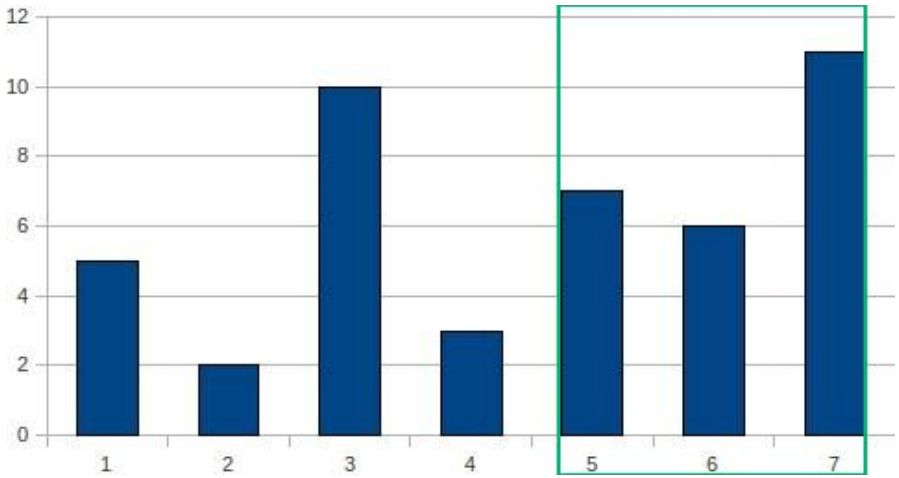
$$< 5, 2, 10, 3, 7, 6, 11 >$$



Consulta: $(t = 2, d = 5)$
Resposta: $\{(5, 10), (2, 10)\}$

Exemplo - Consulta Todos os Pares

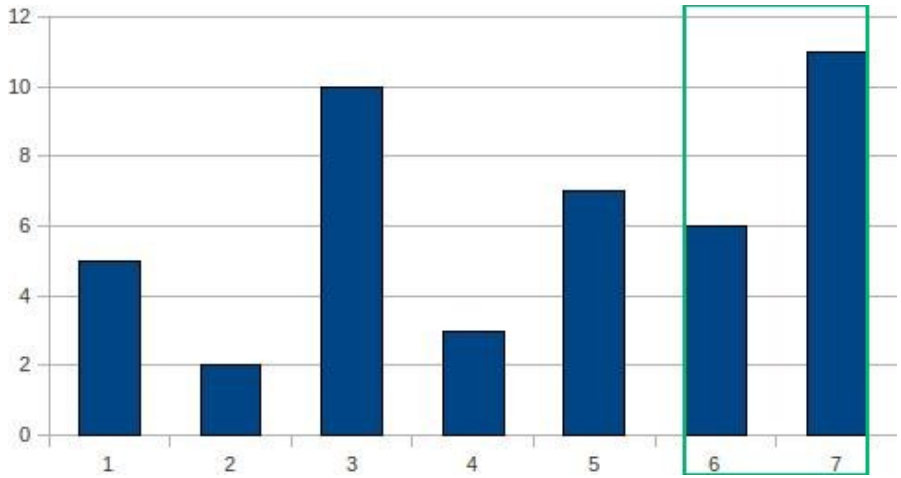
$$< 5, 2, 10, 3, 7, 6, 11 >$$



Consulta: $(t = 2, d = 5)$
Resposta: $\{(5, 10), (2, 10)\}$

Exemplo - Consulta Todos os Pares

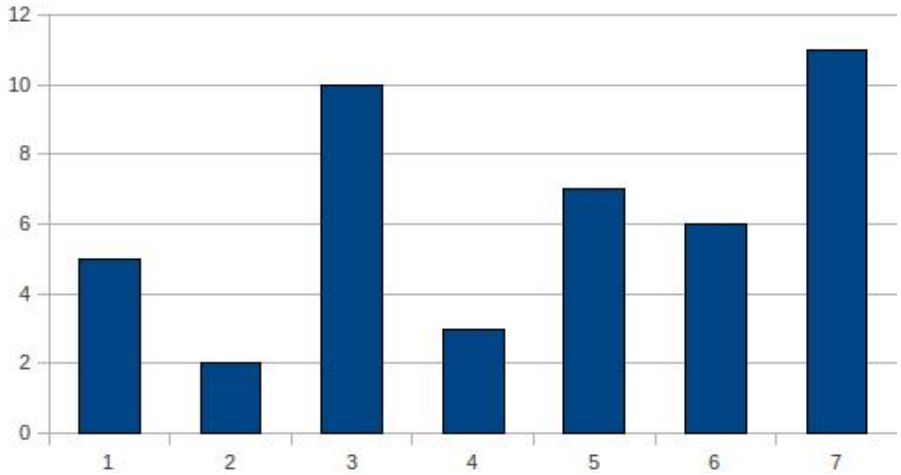
$$< 5, 2, 10, 3, 7, 6, 11 >$$



Consulta: $(t = 2, d = 5)$
Resposta: $\{(5, 10), (2, 10), (6, 11)\}$

Exemplo - Consulta Inícios

$$< 5, 2, 10, 3, 7, 6, 11 >$$



Consulta: $(t = 2, d = 5)$
Resposta: $\{5, 2, 6\}$

Exemplos

$$< 5, 2, 10, 3, 7, 6, 11 >$$

Todos os Pares

Consulta: $(t = 2, d = 5)$

Resposta: $\{(5, 10), (2, 10), (6, 11)\}$

Tamanho da saída varia de 0 *até* $\frac{n(n-1)}{2}$

Inícios

Consulta: $(t = 2, d = 5)$

Resposta: $\{5, 2, 6\}$

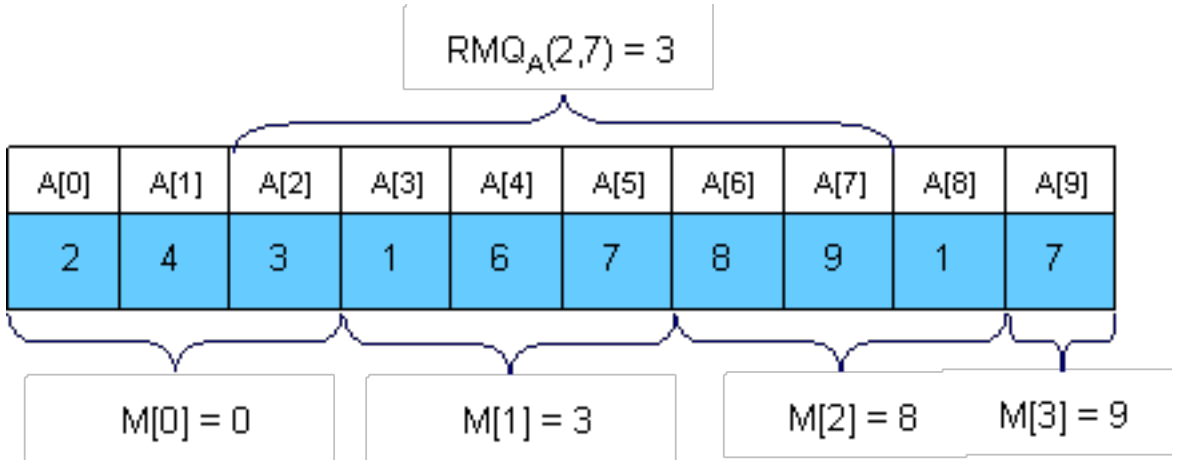
Tamanho da saída varia de 0 *até* n

Objetivos

- Construir estruturas de dados que permitam responder eficientemente os dois tipos de consultas definidas
- Realizar análises teóricas sobre as estruturas propostas
- Avaliar experimentalmente as estruturas

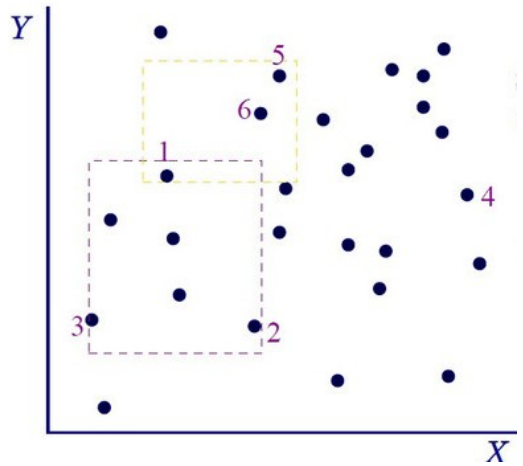
Conceitos Importantes

- Range Minimum Query (RMQ): Determinar o mínimo em um intervalo de forma eficiente. Admite estruturas de dados com as características abaixo:
  - Tempo e espaço de pré-processamento $O(n)$
  - Consultas em $O(1)$



Conceitos Importantes

- Orthogonal Range Tree: Estrutura de dados que permite reportar todos os pontos (em um plano) dentro de um dado retângulo de forma eficiente
  - Tempo e espaço de pré-processamento $O(n \log n)$
  - Consultas em $O(\log n + k)$



Solução Trivial - Todos os Pares

- Pré-processamento: Para todo par (a_i, a_j), $i < j$, guardar as diferenças $a_j - a_i$ em um vetor ordenado
- Consulta: Percorrer o vetor ordenado enquanto as diferenças forem menores que d
- Complexidade de Tempo: $< O(n^2 \log n), O(k) >$, onde k é a quantidade de pares reportados
- Complexidade de Espaço: $O(n^2)$

Solução Trivial - Inícios

- Pré-processamento: Criar uma estrutura de RMQ para a série
- Consulta: Iterar em cada elemento da série e descobrir o maior entre os próximos t elementos
- Complexidade de Tempo: $< O(n), O(n) >$, usando uma estrutura eficiente de RMQ
- Complexidade de Espaço: $O(n)$

Para $k << n$ o tempo de resposta é dominado por n

F-pairs

Um **F-pair** é um par (i, j), tal que:

$$a_i < \min\{a_{i+1}, a_{i+2}, \ldots, a_j\} \text{ e } a_j > \max\{a_i, a_{i+1}, \ldots, a_{j-1}\}$$

Exemplo

$$< 5, 2, 10, 3, 7, 6, 11 >$$

F-pairs: $\{(2, 10), (3, 7), (6, 11), (3, 11)\}$

F-pairs - Propriedades

Lema

Para toda solução (a_i, a_j), temos:

1. (a_i, a_j) *é um F-pair; ou*
2. (a'_i, a_j) *é solução, onde* a'_i *é o início de algum F-pair; ou*
3. (a_i, a'_j) *é solução, onde* a'_j *é o fim de algum F-pair*

Consequência

É possível encontrar todas as soluções, para o problema Todos os Pares, a partir dos F-pairs.

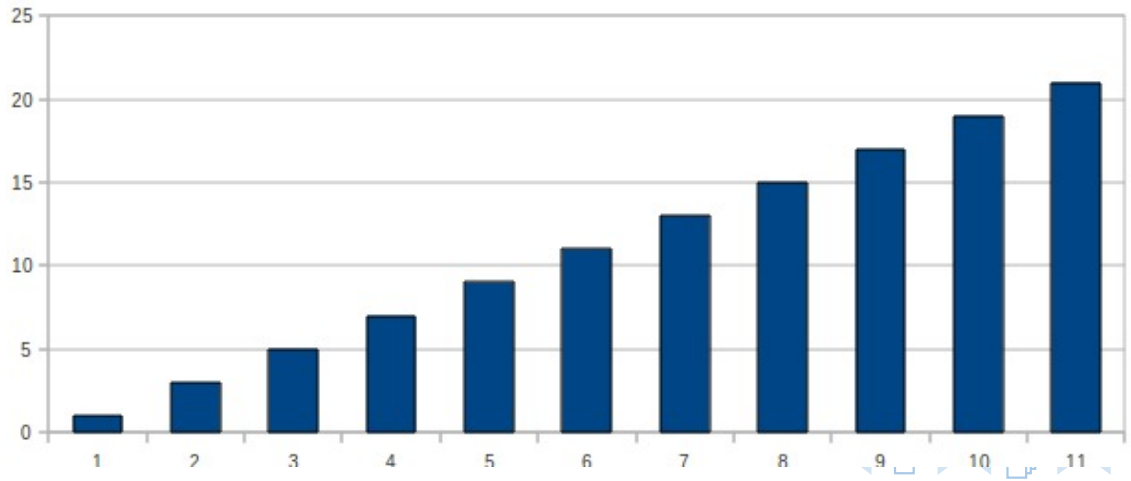
F-pairs - Propriedades

Do ponto de vista da quantidade de F-pairs, toda série pode ser mapeada em uma permutação

Lema

O valor esperado do número de F-pairs é $O(n)$, com constante baixa e o resultado vale com alta probabilidade

Observação: No pior caso, a quantidade de F-pairs pode ser quadrática.



Abordagem Proposta - Todos os Pares

Abordagem

- Pré-processamento: guardar todos os F-pairs em uma estrutura adequada
- Usar os F-pairs como ponto de partida para responder as consultas

Características da Abordagem

- Tempo de resposta $O(k)$
- Tempo de pré-processamento $O(\#\text{F-pairs})$
- Tamanho da estrutura $O(\#\text{F-pairs})$

Análise Experimental

- Avaliar o tamanho das estruturas propostas em séries aleatórias e reais
- Avaliar o tempo de resposta das estruturas
- Comparar com outras estruturas

Análise Experimental - Dados

- Séries geradas aleatoriamente
  - Permutações de tamanho $1MB$, $2MB$, $3MB$, $4MB$ e $5MB$
  - Geradas de forma aleatória
- Séries Reais
  - 4 séries reais advindas de dados do mercado financeiro
  - Tamanhos das séries $199258, 179443, 142941, 191672$

Análise Experimental - Baselines

- RMQ
  - Tempo de resposta $O(n + k)$ para ambas as consultas
  - Tamanho $O(n)$

Cronograma

Novembro	Desenvolver e implementar heurísticas
Dezembro	Desenvolver e implementar heurísticas
Janeiro	Experimentar com séries reais e aleatórias
Fevereiro	Escrever texto
Março	Escrever texto

Referências

Bernard Chazelle.
Functional approach to data structures and its use in multidimensional searching.
SIAM J. Comput., 17:427–462, June 1988.

Johannes Fischer and Volker Heun.
Theoretical and practical improvements on the rmq-problem, with applications to lca and lce.
In *PROC. CPM. VOLUME 4009 OF LNCS*, pages 36–48. Springer, 2006.

F. P. Preparata and M. I. Shamos.
Computational Geometry: An Introduction.
Springer-Verlag, 1985.

Qingmin Shi and Joseph JaJa.
A new framework for addressing temporal range queries and