



## **Chapter VII**

# **Spatial Joins: Algorithms, Cost Models and Optimization**

Nikos Mamoulis, University of Hong Kong, Hong Kong

Yannis Theodoridis, University of Piraeus, Greece

Dimitris Papadias,  
Hong Kong University of Science and Technology, Hong Kong

## **Abstract**

---

*This chapter describes algorithms, cost models and optimization techniques for spatial joins. Joins are among the most common queries in Spatial Database Management Systems. Due to their importance and high processing cost, a number of algorithms have been proposed covering all possible cases of indexed and non-indexed inputs. We first describe some popular methods for processing binary spatial joins and provide models for selectivity and cost estimation. Then, we discuss evaluation of multiway*

*spatial joins by integrating binary algorithms and synchronous tree traversal. Going one step further, we show how analytical models can be used to combine the various join operators in optimal evaluation plans. The chapter can serve as a comprehensive reference text to the researcher who wants to learn about this important spatial query operator and to the developer who wants to include spatial query processing modules in a Database System.*

## Introduction

---

Spatial database systems (Güting, 1994) manage large collections of multidimensional data which, apart from conventional features, include special characteristics such as position and extent. That there is no total ordering of objects in space that preserves proximity renders conventional indexes, such as B<sup>+</sup>-trees, inapplicable to spatial databases. As a result, a number of *spatial access methods* have been proposed (Gaede & Günther, 1998). A very popular method, used in several commercial systems (for example, Informix and Oracle), is the R-tree (Guttman, 1994), which can be thought of as an extension of B<sup>+</sup>-tree in multi-dimensional space. R-trees index object approximations, usually minimum bounding rectangles (MBRs), providing a fast *filter step* that excludes all objects that cannot satisfy a query. A subsequent *refinement step* uses the geometry of the candidate objects (that is, the output of the filter step) to dismiss false hits and retrieve the actual solutions. The R-tree and its variations have been applied to efficiently answer several query types, including spatial selections, nearest neighbors and spatial joins.

As in relational databases, joins play an important role in effective spatial query processing. A *binary* (that is, *pairwise*) spatial join combines two datasets with respect to a spatial predicate (usually *overlap/intersect*). A typical example is “find all pairs of cities and rivers that intersect.” For instance, in Figure 1 the result of the join between the set of cities  $\{c_1, c_2, c_3, c_4, c_5\}$  and rivers  $\{r_1, r_2\}$ , is  $\{(r_1, c_1), (r_2, c_2), (r_2, c_5)\}$ .

The query in this example is a spatial *intersection* join. In the general case, the join predicate could be a combination of *topological*, *directional* and *distance* spatial relations. Apart from the intersection join, variants of the *distance* join have received considerable attention because they find application in data analysis tasks (for example, data mining and clustering). Given two sets  $R$  and  $S$  of spatial objects (or multidimensional points) and a distance function  $dist()$ , the  $\epsilon$ -*distance* join (or else *similarity* join) (Koudas & Sevcik, 2000) returns the

29 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage: [www.igi-global.com/chapter/spatial-joins-algorithms-cost-models/29663](http://www.igi-global.com/chapter/spatial-joins-algorithms-cost-models/29663)

## Related Content

---

### One-Factor Cancellable Fingerprint Template Protection Based on Index Self-Encoding

Yalan Feng, Huabin Wang, Dailei Zhang, Jiahao Liand Liang Tao (2023). *Journal of Database Management* (pp. 1-18).

[www.irma-international.org/article/one-factor-cancellable-fingerprint-template-protection-based-on-index-self-encoding/321546](http://www.irma-international.org/article/one-factor-cancellable-fingerprint-template-protection-based-on-index-self-encoding/321546)

### From 'Flow' to 'Database': A Comparative Study of the Uses of Traditional and Internet Television in Estonia

Ravio Suni (2009). *Database Technologies: Concepts, Methodologies, Tools, and Applications* (pp. 1477-1489).

[www.irma-international.org/chapter/flow-database-comparative-study-uses/7987](http://www.irma-international.org/chapter/flow-database-comparative-study-uses/7987)

### Analysis of National ICT Strategies for USA, EU, India, South East Asia, and Australasia

(2019). *Information Systems Strategic Planning for Public Service Delivery in the Digital Era* (pp. 20-63).

[www.irma-international.org/chapter/analysis-of-national-ict-strategies-for-usa-eu-india-south-east-asia-and-australasia/233401](http://www.irma-international.org/chapter/analysis-of-national-ict-strategies-for-usa-eu-india-south-east-asia-and-australasia/233401)

### Making Query Coding in SQL Easier by Implementing the SQL Divide Keyword: An Experimental Query Rewriter in Java

Eric Draken, Shang Gaoand Reda Alhajj (2011). *Advanced Database Query Systems: Techniques, Applications and Technologies* (pp. 287-303).

[www.irma-international.org/chapter/making-query-coding-sql-easier/52306](http://www.irma-international.org/chapter/making-query-coding-sql-easier/52306)

### Semi-Supervised Event Extraction Incorporated With Topic Event Frame

Gongqing Wu, Zhuochun Miao, Shengjie Hu, Yinghuan Wang, Zan Zhangand Xianyu Bao (2023). *Journal of Database Management* (pp. 1-26).

[www.irma-international.org/article/semi-supervised-event-extraction-incorporated-with-topic-event-frame/318453](http://www.irma-international.org/article/semi-supervised-event-extraction-incorporated-with-topic-event-frame/318453)