CROSS-BORDER TOPLOGICAL JOIN OPTIMIZATION OF DISTRIBUTED SPATIAL DATA BASED ON ZONAL FRAGMENTATION

ZHU Xinyan[a] *, ZHOU Chunhui[a], CHEN Di[a]

[a] Liesmars, Wuhan University, Luoyu Road, 430079 Wuhan , China

Commission IV

KEY WORDS: Spatial Database, Zonal Fragmentation, Cross-Border, Topological Join, Distributed Query, Optimization

ABSTRACT:

Spatial data fragmentation classifies in zonal fragmentation and layer fragmentation in distributed Spatial database. Because of the geospatial continuity and strong correlation between spatial data, cross-border query becomes an inherent problem in distributed spatial query based on zonal fragmentation, and cross-border fragment join optimization is a core issue. Firstly, this paper discussed the general mean of grouping of the fragment joins, and they are divided into to groups, NCBJs and CBJs; Secondly the spatial topological predicates further the spatial joins are classified; then the optimization of 4 class of CBJs are discussed in detail, and the removing, filtering, transforming rules are proposed, further the processing algorithms. Tests are designed to examine the proposed methods, and the results show that the proposed methods improve the efficiency of cross-border join greatly.

1. INTRODUCTION

Spatial database is the core of Geographic Information System (GIS). The distributed characteristic of geospatial data in its production, management, maintenance and applications causes the spatial data management inevitably moving towards a distributed way. The fragmentation in a distributed spatial database can be classified as Zonal Fragmentation (ZF) and Layer Fragmentation (LF) (R. Laurini, 1998). ZF (also named as spatial partitioning, or horizontal fragmentation) means that spatial data for the whole geographical coverage are split into several homogeneous database tables on the basis of regions and stored in different sites; LF (also named thematic fragmentation, or vertical fragmentation) means that spatial data for the same geographical coverage are deposited in layers by their themes and stored in different sites. In fact, data can also be fragmented and distributed through a mix method of both. At present, researches on spatial join optimization are mostly concentrated in LF circumstances, and rarely in ZF. This paper proposes a topological join optimization of distributed spatial data based on ZF.

In ZF, distributed spatial data management has its own characteristics, which are cross-border spatial correlation (CBSC) and cross-border seamless query (CBSQ) issues.as shown in Fig.1, when selecting features that touch with object *a1* stored in Site A, it will involve *b1* and *b3* stored in Site B in Fig.1(a). When selecting features that within the buffer of line feature L, it may involve some features stored in Site A in Fig.1(b). In addition, the buffering query will involve some different non-adjacent spatial fragments only if the buffer radius is large enough. According to the join allocation regulations in traditional databases (D. Kossmann, 2000; M. T. Ozsu and P.Valduriez, 2002), one of seamless query methods of the distributed spatial data is translating the whole join operation into the joins between each fragment. Meanwhile, there may be a large number of ineffective or invalid spatial fragment joins, which not only increase fragment join time, but also increase the amount of data transmitting between different sites, so the query processing efficiency will be greatly cut down. So the key of the problem is how to optimize spatial fragment joins in a cross-border query.

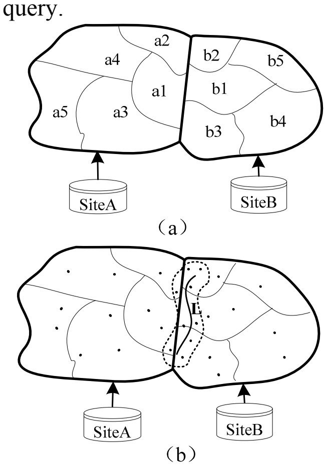


Fig.1 the problem of cross-border query in distributed spatial database based on zonal fragmentation

There are many research results in spatial fragment join optimization. Jacox and Samet comprehensively summed up the studies on spatial join technology (E. H. Jacox and H. Samet, 2007). Cross-border join optimization of distributed spatial data based on ZF is seldom concerned. The existing spatial join optimization methods are insufficient: (1) Studies on spatial join predicates are not comprehensive — most studies use

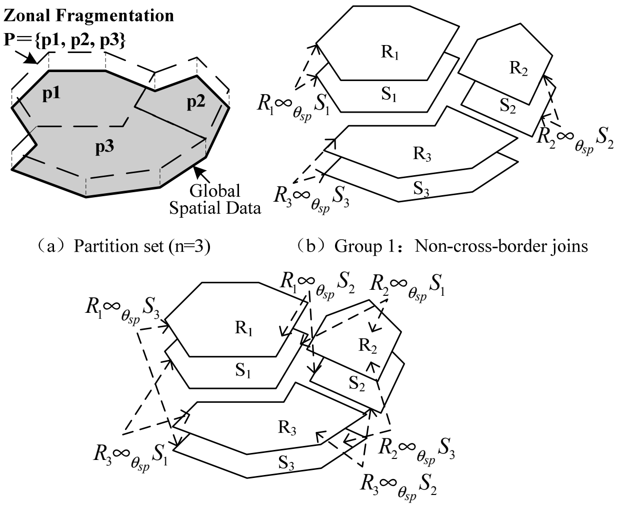
* ZHU Xinyan, PhD, Professor, majors in spatial database, web GIS, spatial statistics etc. Email: zxy@lmars.whu.edu.cn.

Intersects while its join optimizations are not applicable to all spatial topological predicates; (2) Many existing spatial join optimization methods cannot be applied to cross-border joins in a distributed spatial database based on ZF. At present, there are three dominant strategies of spatial join operations of the *Intersects* predicate in distributed spatial query processing: Naive strategy (D. J. Abel, 1995), Semi-join strategy (W. G. Aref, 1997) and MR2(Multiple step with Remote indices, Version 2)(M. R. Ramirez and J.M. Souza, 2001). However, the above spatial join strategies do not take the specificity of ZF into consideration.

2. CROSS-BORDER TOPOLOGICAL JOIN OPTIMIZATION OF DISTRIBUTED SPATIAL DATA

2.1 Cross-border join and Non-Cross-border join base on ZF

Assume that the global spatial relationships R and S are partitioned seamlessly by n polygons, P = { p1、p2、…、pn}, and the results are two Fragment Set, also named Partition Set (PS): PS(R)={R_1，R_2，…，R_n}, PS(S) ={S_1，S_2，…，S_n}.


Zonal Fragmentation
P={p1, p2, p3}
p1
p2
p3
Global Spatial Data
R1
S1
R2
S2
R3
S3
$R_1\infty_{\theta_{sp}}S_1$
$R_2\infty_{\theta_{sp}}S_2$
$R_3\infty_{\theta_{sp}}S_3$
$R_1\infty_{\theta_{sp}}S_2$
$R_2\infty_{\theta_{sp}}S_1$
$R_1\infty_{\theta_{sp}}S_3$
$R_3\infty_{\theta_{sp}}S_1$
$R_2\infty_{\theta_{sp}}S_3$
$R_3\infty_{\theta_{sp}}S_2$

（a）Partition set (n=3)　　（b）Group 1：Non-cross-border joins

（c）Group 2：Cross-border joins

Figure 2. Translating a global join to fragment-fragment joins

As shown in Fig.2(a) with n=3, the global join is translated into 9 fragment joins, and these fragment joins can be divided into the following two groups: (1) Group 1 is Non-Cross-Border Joins (NCBJs). Two fragments in each fragment join have the same spatial extent, as shown in Fig.2(b). (2) Group 2 is Cross-Border Join (CBJs). Two fragments in each join are not overlapping, as shown in Fig.2(c). Generally, groups are shown as formula (1).

$$\text{Group 1:}\quad (R_1\infty_{\theta_{sp}}S_1),\quad (R_2\infty_{\theta_{sp}}S_2),\ldots,(R_n\infty_{\theta_{sp}}S_n)$$

$$\text{Group 2:}\quad (R_1\infty_{\theta_{sp}}S_2),\ldots,(R_1\infty_{\theta_{sp}}S_n),(R_2\infty_{\theta_{sp}}S_1),(R_2\infty_{\theta_{sp}}S_3),\ldots,$$
$$,(R_2\infty_{\theta_{sp}}S_n),\ldots,(R_n\infty_{\theta_{sp}}S_1),\ (R_n\infty_{\theta_{sp}}S_2),(R_n\infty_{\theta_{sp}}S_{n-1})$$

where ∞ denotes a join operation, and θ_{sp} denotes spatial join predicate.Obviously, the number of CBJs is more than the number of NCBJs, and with the increasing number of zonal fragments, CBJs is also rapidly growing. So the optimization of CBJs is crucial for a spatial join based on ZF.

2.2 The classification of spatial topological relationship predicates

OGC's SFA (Simple Feature Access) specification defines the spatial topological predicates based on 4IM (Clementini, E. and P.D. Felice, 1996) and Dimensionally Extended Nine-Intersection Model (DE-9IM) (Clementin and P.D. Felicei, 1995). Eight common spatial topological predicates are defined by OGC SFA, including *Crosses*, *Disjoint*, *Within*, *Contains*, *Equals*, *Touches*, *Intersects*, *Overlap.* According to the definition, the eight spatial topological predicates make up a complete topological relation set. In this set, *Disjoint* is mutually exclusive with other predicates. So all the spatial topological relations denoted by these predicates can be divided into two classes, as shown in Table 1. Only CBJs of two adjacent fragments with the 1st class predicates are meaningful for the final result, while all CBJs with the predicate *Disjoint* are valid for the result.

Table 1. Classification of spatial topological relations

Classification	Spatial topological relationship predicates
1st class	Crosses、Within、Contains、Equals、 Touches、Intersects、Overlaps
2nd class	Disjoint

2.3 The classification of spatial topological joins

Spatial topological join is often used in combination with spatial analysis operations(*Buffer*, *Distance*, *Intersection*, *ConvexHull*, for example). And *Buffer* is a specific spatial analysis predicate. So the spatial join optimization should be specially considered when it come to *Buffer* predicate. Under given conditions *Distance* and *Buffer* predicates can convert each other. Therefore, this paper takes the particularity of *Buffer* operation in spatial join into consideration. The spatial topological joins can be divided into four classifications: (1)topological *Intersects* join(TIJ), to denote the spatial join based on the 1st class topological predicates. (2)topological *Disjoint* join(TDJ), to denote the spatial join based on 2nd class topological predicates. (3) hybrid spatial *Intersects* join(HSIJ), to denote the combined spatial join based on the 1st class topological predicates and *Buffer* predicates.(4)hybrid spatial *Disjoint* join(HSDJ), to denoted the combined spatial join based on the 2nd class topological predicates and *Buffer* predicate. For example, "which counties are crossed by the highway ?" is a TDJ query; "which counties are within the 5km buffer of the highway?" is a HSIJ query; "which counties are outside the 5km buffer of the highway?" is a HSDJ query.

2.4 Optimization of cross-border topological join

CBJs are also divided into 4 classes, and the optimization is considered according to the classification.

Definition 1. Intersecting Rectangle(IR): For two fragment A and B involved in join, their own MBRs are denoted by MBR(A) and MBR(B). So the intersection of MBR(A) and MBR(B) is MBR(A)∩MBR(B) which is called the intersecting rectangle of fragments.

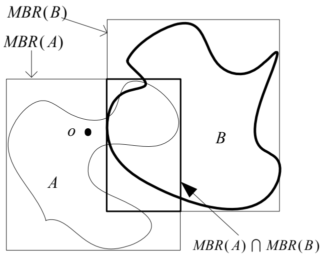


Figure 3. The Intersecting Rectangle(IR) of 1st class CBJs

As shown in Fig.3, IR can be used to the optimization of 1st and 2nd class CBJs. Obviously, the object o in fragment A does not intersect with IR, as the same in fragment B.Given the theorem 1.

Theorem 1: For any two fragments in ZF: if there are objects in two fragments which are to meet the 1st class spatial topological relationship, the objects are bound to intersect the IR.

To omit the proof here. According the theorem 1 the 1st class and 2nd CBJs can be optimized. As followed Rule 1, the removal rule of 1st class CBJs has been given to remove the redundant objects which will not involved in final results.

Rule 1. The removal rule of the 1st class CBJs: if there is no intersection of the MBRs of two spatial fragments, the 1st class CBJ can be removed.

As followed Rule 2, the Filtering rule of the 1st class CBJs has been given, and the efficiency of CPU computing and the data transmission get higher by using the IR to filter the redundant objects.

Rule 2. Filtering rule of the 1st class CBJs: Using the IR to filter the spatial objects of two fragments, then performing the spatial join operation on the filtered results.

The 2nd class CBJs means those CBJs with a predicates *Disjoint*. For 2nd class CBJs，if there is no intersection of the MBRs of two fragments, that is the IR is NULL, all spatial objects in the two fragments are to meet the *Disjoint* relationship. So the result of the CBJ is just their Cartesian Product. Because this alternative is not to compute the complex spatial relationships, it will play a role in optimization. Rule 3 gives the strategy.

Rule 3. Transformation rule of the 2nd class CBJs: If IR is NULL, the 2nd class CBJs can be replaced with their Cartesian Product.

Note： If the IR of a 2nd class CBJ is not NULL, the two fragments involved in the CBJ can not be simply filtered by the IR. It can be performed with the method of traditional spatial join in centralized spatial database

If involved *Buffer* predicates in spatial topological join, there is a complex case, and each CBJ is a hybrid spatial join. For optimizing this complex case, we defined the Expanded Rectangle(ER) and the Expanded Intersecting Rectangle(EIR).

Definition 2. Expanded Rectangle(ER): Assume the bounding coordinates of rectangle R in the four directions are (XL, YL, XH, YH). Then, to expanding the rectangle with a certain distance d in the four direction obtained a expanded rectangle R'(XL-d, YL-d, XH+d, YH+d). It is called expanded rectangle of rectangle R, and denoted by R-E(d), where d is the expanding distance or the buffer radius.

Definition 3. Expanded Intersecting Rectangle(EIR) of fragments: For two fragments A and B involved in hybrid spatial join, their own MBR are denoted by MBR(A) and MBR(B). The Expanded rectangle of their MBRs are denoted by MBR(A)-E(d) and MBR(B)-E(d). And the Expanded intersecting rectangle of fragments A and B is denoted by MBR(A)-E(d)∩MBR(B)-E(d), where d is the expanding distance or the buffer radius.

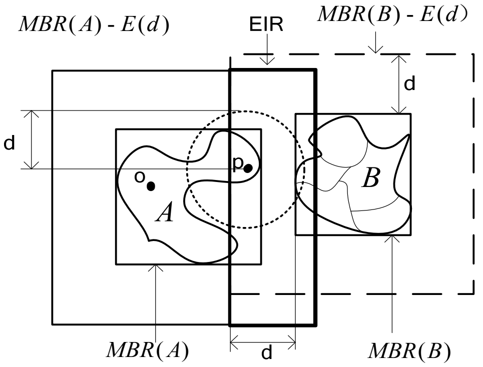


Figure 4. The Expanded Intersecting Rectangle(EIR) of 3rd class CBJs

As shown in Fig.4, it concluded that the 3rd class hybrid join (HSIJ) can be optimized using EIR. Obviously, the object *o* in fragment A dose not intersect with EIR, so the *Buffer* of *o* with a buffer radius d, does not intersect with any objects of fragment B. While the *Buffer* of object *p* intersects with some objects in fragment B, then *p* must intersect with the EIR. The theorem 2 is given here.

Theorem 2: For any two fragments in ZF: if there are objects in two fragments which are to meet the 1st class spatial topological relationship with a *Buffer* operation, the objects are bound to intersect the EIR, the *Buffer* parameter d is just the parameter of the EIR.

According the theroem 2, the removal rule (Rule 4) and the filtering rule (Rule 5) of the 3rd class CBJs can be obtained.

Rule 4. The removal rule of the 3rd class CBJs: if there EIR of two spatial fragments is NULL, the 3rd class CBJ can be removed.

Rule 5. The Filtering rule of the 3rd class CBJs: Using the EIR to filter the spatial objects of two fragments, then performing the spatial join operation on the filtered results.

Rule 4 can removal the redundant spatial join, and rule 5 is to reduce the number of redundant object in fragments to achieve the optimization of spatial join.

For 4th class CBJs, if EIR is NULL, so the optimizating method is the same with 2nd class CBJs, and the spatial join can be replaced by the Cartesian Product of two fragments

2.5 The processing algorithms of the CBJs in distributed environment

For the CBJs based on ZF, it is critical to perform the optimization by the classification. According to the above analysis, in the distributed environment, the processing of the 1st or 3rd class CBJs can be divided into 3 phases: Firstly, calculating the the IR or EIR two spatial fragment A and fragment B involved in spatial join, if the IR or EIR is NULL, the spatial join relating to these fragments can be removed. Secondly, the interim result set A' can be generated by filtering the fragment A with IR or EIR. Finally, the set A' would be sent to the site stored the fragment B. Then to perform the spatial join.

For the 2nd or 4th class CBJs, their processing can be also divided into 3 phases: Firstly, calculating the IR or EIR two spatial fragment A and fragment B involved in spatial join, if the IR or EIR is NULL to go to the second step, else to the third step. Secondly, the fragment A (not including the geometry data) would be sent to the site stored fragment B. Calculating the Cartesian Product of two fragments. the end. Finally, the fragment A(including the geometry data) would be sent to the site stored fragment B. Then to perform the spatial join with fragment B get the final result.

3. TEST AND ANALYSIS

3.1 Test environment and data set

Tests are carried out on PC nodes (CPU P4 2.4G, Memory 2GBytes) in a LAN. There are two test data sets which come from the fundamental geography data of China. Data set 1 (Counties) are the administrative area boundaries of all cities and counties in four provinces (Hubei Province, Jiangxi Province, Hunan Province and Jiangsu province), and they are partitioned to four parts by the administrative boundaries of the 4 provinces; Data set 2 (Highways) are the highways in the same 4 provinces; the scale of data is 1:250,000. And the test data is stored in 4 nodes separately according to their concerned administrative area. Total data size is about 12MBs. The databases is Oracle Spatial 10g. And we also use Internet Communication Engine (ICE) for communication. Visual studio C++ 2005 is the main development tool.

This paper designs 3 test to check the optimizing strategies. The first is to test and analysis of the IR Filtering Optimization Strategy (IFOS) of the 1st class CBJs, and the second is to test and analysis the Cartesian Product Transforming Strategy (CPTS) of 2nd class CBJs. The third is to test and analysis the EIR Filtering Optimization Strategy (EFOS) of the 3rd class CBJs. Three spatial join strategies are compared here: Naive strategy (NS), Semi-join strategy(SS) and the Optimization Strategy of CBJs (OSC, including IFOS, CPTS, and EFOS). Because adaptability of MR2 are restricted, it is not included.

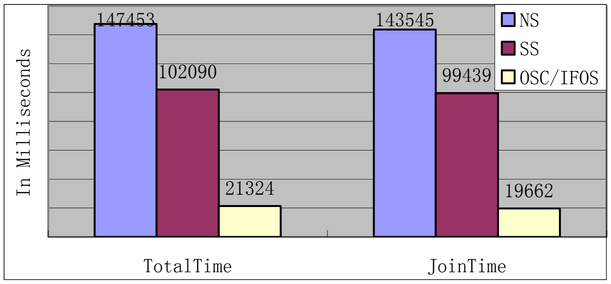
3.2 Test and analysis of the 1st class CBJ's performance based on adjacent fragments in ZF

The test is used to compare the efficiency of NS, SS and IFOS. We designed above mentioned three kinds of performing strategies for CBJs. The query is:

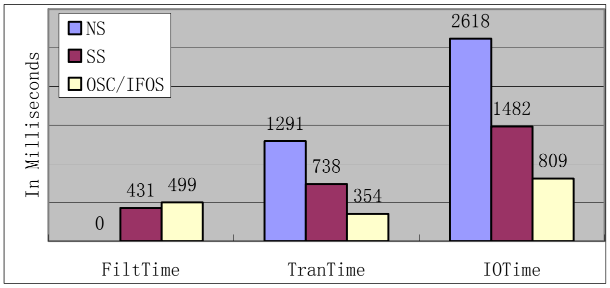
Query 1: Select a.name, b.name from Counties a, Counties b where Touches(a.shape, b.shape) = 'TRUE'；

During fragmentation, the table *Counties* is partitioned into 3 fragments (*Hubei_counties*, *Jiangxi_counties*, and *Hunan_counties*). According to the fragmentation of the data set, there will be 3 CBJs (*Hubei_counties* ↔ *Jiangxi_counties, Hubei_counties* ↔ *Hunan_counties, Jiangxi_counties* ↔ *Hunan_counties*), and each CBJ is tested using three spatial join strategies including NS, SS, OSC/IFOS.

For the same test data, we can obtain the same query results through the 3 strategies, and the correctness of the strategies has been checked. The cost time of every stages of spatial join is the average value of the testing results of 3 CBJs. The mentioned 3 strategies are compared, as shown in Fig.5. The comparison of total time, join time is shown in Fig.5(a); The comparison of filtering time, transmitting time and I/O time (including the index constructing and the database writing) is shown in Fig.5(b).



(a)



(b)

Figure 5. Cost comparison of 3 join strategies with the 1st class CBJs based on ZF (in milliseconds)

From the result, it can be drawn that the total time of OSC/IFOS strategy is minimal, and followed by SS strategy, and NS strategy is maximal. Meanwhile, the optimization of OSC/IFOS strategy is efficient obviously using the filter operation before spatial join, which can cut down the time-consuming of transmission, constructing index and local connection.

3.3 Test and analysis of the 2nd class CBJs' performance

In ZF, if the IR is not NULL, the 2nd class CBJs can not be optimized. We designed this test to test the optimization of OSC/CPTS when the IR is NULL. The 2nd class CBJs can not be optimized by SS strategy, so only NS strategy and OSC/CPTS strategy mentioned above is used in this test. We use the fragments *Hubei_counties* and *Jiangsu_counties* to make up a 2nd class CBJ, and the IR is NULL. Query 2 and 3 can depict the difference of two strategies.

Query 2 (Using NS strategy): Select A.name, B.name from Hubei_counties A, Jiangsu_counties B where Disjoint (A.shape, B.shape) = 'TRUE'；

Query 3 (Using OSC/CPTS **strategy):** Select A.name, B.name from Hubei_counties A, Jiangsu_counties B；

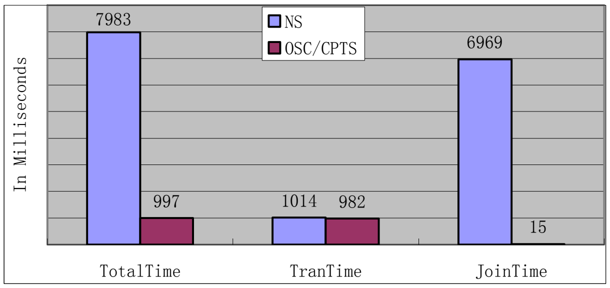


Figure 6. Cost comparison of 2 join strategies with the 2nd t class CBJs based on ZF (in milliseconds)

As shown in Fig.6, for the 2nd class cross-border topological join based on zonal fragmentation, the efficiency of OSC/CPTS strategy is much higher than NS strategy when the IR is NULL.

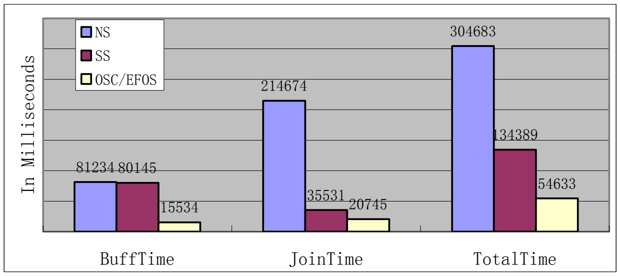
3.4 Test and analysis of the 3rd class CBJs' performance

The 3rd class spatial join is a hybrid join, which involves a 1st class topological predicate and a *Buffer* operation.Query 4 is a 3rd class spatial join.

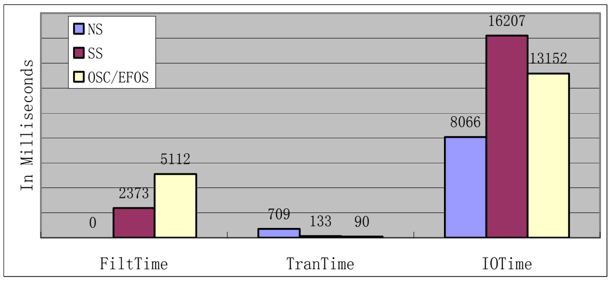
Query 4: SELECT S.name, H.name FROM Counties S, Highways R WHERE *Intersect*(S.shape, *Buffer*(H.shape, 10, "unit=KM") = 'TRUE'；

During fragmentation, the table *Countie*s and table *Highways* are both partitioned into 3 fragments (Hubei area, Jiangxi area, and Hunan area). According to the fragmentation, there are 6 CBJs (*Hubei_counties* ↔ *Jiangxi_highways, Hubei_counties* ↔ *Hunan_highways, Jiangxi_counties* ↔ *Hubei_highways, Jiangxi_counties* ↔ *Hunan_highways, Hunan_counties* ↔ *Hubei_highways and Hunan_counties* ↔ *Jiangxi_highways*), and the others are 3 NCBJs (*Hubei_counties* ↔ *Hubei_highways, Jiangxi_counties* ↔ *Jiangxi_highways, and Hunan_counties* ↔ *Hunan_highways*).

We perform the 6 CBJs with 3 strategies including NS, SS and OSC/ EFOS, and take average costs of every stage of the 6 CBJs. The result is shown in Fig. 7. The comparison of total time, join time and *Buffer* calculating time is shown in Fig. 7(a)., and the comparison of filtering time, transmitting time and I/O time (including the index constructing and the database writing) is shown in Fig.7(b). It is obvious that the join time and the *Buffer* calculating time hold the main part of the total time. NS does not make any efforts of filtering according to a distributed environment based on ZF, so it costs much more than the other two strategies. SS uses the MBRs in place of the objects themselves to participate in the join step, so it can reduce the join cost and the transmitting cost; But it does not consider the specification of a hybrid join, it can not reduce the *Buffer* operation's calculation. OSC/EFOS gives full consideration to both the topological predicates and the *Buffer* operation, through a lightweight filtering step, OSC/EFOS can reduce the *Buffer* operation's calculation, the join calculation, and the objects needed to be transmitted. So, OSC/EFOS can perform the 3rd CBJs effectively.



(a)



(b)

Figure 7. Cost comparison of 3 join strategies with the 3rd class CBJs based on ZF (in milliseconds)

4. CONCLUSION

Cross-border spatial query is an inherent problem in distributed spatial database based on Zonal Fragmentation. And the optimization of such CBJs is one of key strategies which can improve the efficiency of distributed spatial query. We discussed the optimization of the distributed spatial query based on a Zonal Fragmentation. The theorems, rules and methodologies are proposed in this paper. Through the grouping, classification, different types of CBJs are given corresponding optimization strategies. Test results have shown the effectiveness of the proposed methods.

REFERENCE：

Clementini, E., P.D. Felice, and P.v. Oosterom, 1993. A Small Set of Formal Topological Relationships Suitable for End-User Interaction, in Proceedings of the Third International Symposium on Advances in Spatial Databases. Springer-Verlag.

Clementini, E. and P. Di Felice, 1995. A comparison of methods for representing topological relationships. *Information Sciences - Applications*, 3(3), pp. 149-178.

Clementini, E. and P.D. Felice, 1996. A model for representing topological relationships between complex geometric features in spatial databases. *Inf. Sci.*, 90(1-4), pp. 121-136.

D. J. Abel, et al., 1995. Spatial Join Strategies in Distributed Spatial DBMS, *in Proceedings of the 4th International Symposium on Advances in Spatial Databases*. Springer-Verlag. pp. 348-367.

D. Kossmann, 2000. The state of the art in distributed query processing. *ACM Computing Surveys (CSUR)*, 32(4), pp. 422-469.

E. Clementini, J. Sharma, and M.J. Egenhofer, 1994. Modelling topological spatial relations: Strategies for query processing. *Computers and Graphics*, 18(6), pp. 815-822.

E. H. Jacox and H. Samet, 2007. Spatial join techniques. *Acm Transactions on Database Systems*, 32(1), pp. 44.

J.R. Chen, J.R. Yan, T.R. Ye., 1992. Introduction to Distributed Database Design. Tsinghua press,

K. L. Tan, B. C. Ooi, and D.J. Abel, 2000. Exploiting spatial indexes for semijoin-based join processing in distributed spatial databases. *IEEE Transactions on Knowledge and Data Engineering*, 12(6), pp. 920-937.

M. R. Ramirez and J.M. Souza, 2001, Distributed Processing of Spatial Join. *Tese DSc*, *Coppe/UFRJ*, pp. 8.

M. T. Ozsu and P.Valduriez, 2002. Principles of Distributed Database Systems(Second Edition), Tsinghua press.

R. Laurini, 1998. Spatial multi-database topological continuity and indexing: a step towards seamless GIS data interoperability *International Journal of Geographical Information Science*, 12(4), pp. 30.

W. G. Aref, 1997. Query Processing in Distributed Spatial Databases, in First International Conference and Workshop in Interoperating Geographical Information Systems. Santa Barbara, California.