Spatial Join on Positional Uncertain Data

Welder B. Oliveira[1], Sávio S. T. Oliveira[1], Vagner J. S. Rodrigues[1], Helton S. B. Santos[2], Kleber V. Cardoso[1]

[1]Instituto de Informática – Universidade Federal de Goiás (UFG)
Caixa Postal 131 – 74.001-970 – Goiânia – GO – Brazil

[2]Departamento de Estatística – Universidade de Brasília (UNB)
Campus da Unb, Sgan - Asa Norte, UnB - Brasília, DF, 70910-900.

Abstract. *This paper presents a probabilistic spatial join on positional uncertain data designed to be a) generalist; b) accurate and c) efficient. A proposed progressive Monte Carlo algorithm is used in the refinement step and the Chebyshev inequality is applied in the filtering one in order to provide efficiency, efficacy and generality. The experiments show that the current propose is Pareto efficient concerning these requirements, i.e., it is not outperformed by any competing method. Also, the solution's parameters relating accuracy and efficiency may be adjusted to maximize the gain in one while relaxing the other according to user's demand.*

1. Introduction

Spatial applications frequently need to combine two data sets based on some spatial relationship between objects in these two data sets [Patel and DeWitt 2000]. That is the case when a spatial join is required. While very useful, in general, the evaluation of spatial join predicates has high computing costs and demand sophisticated solutions in order to achieve satisfactory performance for large datasets as one can see in [Patel and DeWitt 2000].

The computational cost becomes even higher when we need to deal with objects which the position is not precise. This can happen due to several reasons, e.g., due to the imprecision of the georreferecing technique employed to acquire the data. In this context, the join predicate may not be accurately evaluated, having severe impact on the results presented by the applications. Thus, robust solutions based on some probabilistic approach are natural candidates for improving the spatial join. We indroduce a new algorithm based to perform a probabilistic spatial join based, which we call Progressive Monte Carlo Spatial Join (PMCSJ). Furthermore, PMCSJ is designed to address the following requirements: accuracy, efficiency and generality. Section 2 comments the related works, Section 3 shows the solution, Section 4 presents the results and Section 5 concludes the work.

2. Related Work

There exists several methods for performing a deterministic spatial join, using or not indexed datasets [Mishra and Eich 1992, Arge et al. 1998, Jacox and Samet 2003, Luo et al. 2002, Patel and DeWitt 1996, Elmasri 2008, Lo and Ravishankar 1994, Brinkhoff et al. 1993, Huang et al. 1997]. These methods assume absolute precision in spatial objects' coordinates. Also, several works

[Wolfson et al. 1999, Pfoser and Jensen 1999, Yu and Mehrotra 2003, Zhang et al. 2012] have been proposed to deal with spatial joins on moving objects, which is not the case of the present work since we are interested in modeling the error coming from imprecise georreferencing and not by moving objects. Spatial join on existentially uncertain were explored by [Dai et al. 2005] and [Ljosa and Singh 2008] presented an approach able to deal with both existencial and positional uncertainty by using a score function in which the two kinds of uncertainty are considered.

The work [Openshaw 1989] proposed the use of the Monte Carlo Method (MMC) to compute the intersection probabilities of positional uncertain geometries, which we will call here Random Spatial Join (RSJ). A limitation faced by RSJ is the computational cost of the MCM. However, RSJ is generic method concerning the PDF error assumption and we will compare it with PMCSJ in Section 4.

In the probabilistic spatial join proposed by [Ni et al. 2003], the error is modeled using a Circular Normal distribution. However, as it does not handle errors from others distributions, it will not be tested.

3. Probabilistic Spatial Join

Like the main proposals in deterministic spatial join, PMCSJ is devided in the classical two steps: *filtering* and *refinement*. The filtering step is performed on a novel probabilistic version of the MBR - the confidence rectangles (CR). The refinement is done by the novel Progressive Monte Carlo Method (PMCM).

3.1. Filtering Step

The index is built using an *R-tree* with CRs replacing MBRs. In order to do that, the MBR for a given geometry is expanded in vertical and horizontal directions by a shift value d. A formal definition for CR is presented below.

Definition 1. *Given a geometry g and a threshold probability p, a confidence rectangle (CR) with threshold probability p for g is the one which contains the MBR of g and the probability of g lies inside it is at least equal to $\sqrt{1-p}$.*

Theorem 1. *Let G and H be CRs for the geometries g and h respectively. If G and H do not intersect then the intersection probability between g and h are less than p.*

Proof. Since G and H are CRs for g and h then, follows from the definiton that $Pr(g \subset G) \geq \sqrt{1-p}$ and $Pr(h \subset H) \geq \sqrt{1-p}$. Then, assuming independence in the error direction in the g and h coordinates,

$$Pr(g \subset G, h \subset H) = Pr(g \subset G).Pr(h \subset H) \geq (\sqrt{1-p})(\sqrt{1-p}) = 1-p.$$

Thus, $Pr(g \not\subset G \cup h \not\subset H) < p$.

If G and H do not intersect then g or h must not be contained in its CR, whose probability is less than p by the above equation. Thus, $Pr(g \cap h) < p$. □

To build a CR, we use of the Chebyshev inequality, which declares: If X is a integrable random variable with finite mean $\mu = E(X)$ and standard deviation σ, then for any $k > 0$, $Pr(|X - E(X)| > k\sigma) < 1/k^2$. For X positive,

$$Pr(X - E(X) > k\sigma) < 1/k^2 \implies Pr(X \leq E(X) + k\sigma) \geq \frac{k^2 - 1}{k^2}.$$

In our case, X is the positional error. Thus, the probability for the error be at most $d = E(X) + k\sigma$ is at least $\frac{k^2-1}{k^2}$. As a given geometry must lie in the CR with a probability of at least $\sqrt{1-p}$, then

$$\sqrt{1-p} = \frac{k^2 - 1}{k^2} \implies k = \sqrt{\frac{1}{1 - \sqrt{1-p}}}.$$

The CR is given by the coordinates $P_{min} = (x_{min}, y_{min})$ and $P_{max} = (x_{max}, y_{max})$, with $x_{min} = x_{MBR\_min} - d$, $y_{min} = y_{MBR\_min} - d$, $x_{max} = x_{MBR\_max} + d$ and $y_{max} = y_{MBR\_max} + d$.

3.2. Refinement

The *refinement* step is given by the Algorithm 1 which perform the novel PMCM procedure. The performance gain comes from avoiding more simulations than the sufficient to guarantee the predicate evaluation with a condifence level γ for each candidate geometry pair.

Algorithm 1: Progressive Monte Carlo Method.

Data: a and b: two geometries;
p: threshold probability;
m: size of each simulation batch;
n_{max}: maximum number of simulations allowed
Result: Return TRUE if the success proportion is at least equal to p, FALSE otherwise.

1 Initialize with zero the counter n (number of Monte Carlo realizations up to now).
2 Shift the a and b coordinates m times.
3 Update n by m units ($n \leftarrow n + m$).
4 Compute the proportion $\hat{q}$ of success in the n simulations.
5 Compute the Confidence Interval (CI) for q, i.e.,

$$CI(q, \gamma) = \left[\hat{q} - t_c\sqrt{\frac{\hat{q}(1-\hat{q})}{n}}, \hat{q} + t_c\sqrt{\frac{\hat{q}(1-\hat{q})}{n}}\right]$$

where t_c is the $(1+\gamma)/2$ quantile of the t-Student distribution with $n-1$ degrees of freedom.
6 If $p \in IC(q, \gamma)$ and $n < n_{max}$, go to steps 2-5.
7 If $\hat{q} \geq p$ return **TRUE** else return **FALSE**.

4. Results

4.1. Test setup

The datasets used in test were: 1) vegetation, 2) deforestation and 3) wildfire areas, both from the Brazilian province called Goiás and procuced by the LAPIG-UFG laboratory. The join executed was vegetation against the union of deforestation with wildfire areas. The competing approach are Random Spatial Join (RSJ) with $m = 150$ simulations and PMCSJ with a maximum number of simulations $n.max$ of 150 and 1000. The positional error Y follows a Half Normal distribution, defined by $Y = |X|$, with $X \sim N(200, 100^2)$. The mean and standard deviation (required as parameters for CR computation) of the Half Normal distribution are: $E(X) = \sigma\sqrt{\frac{2}{\pi}}$ and $sd(X) = \sigma\sqrt{1 - \frac{2}{\pi}}$. The threshold probabilities tested are: 0.10, 0.20, ..., 0.80 and 0.90. The accuracy parameter for PMCSJ is $\gamma = 0.99$, the size of each simulation batch set to 50.

A ground truth database was used to compare the methods, which contains the reference intersection probabilities for each geometry pair (a_i, b_j), with $a_i \in A$, $b_j \in B$, with A and B being the two joined data sets. The ground truth was built with a larger number N of Monte Carlo simulations, in our work, $N = 500$. The number $N = 500$ provides a margin or error $E = 0.045$.

The metrics used for comparison are: i) proportion of neighborhood false positives; ii) proportion of neighboorhod false negatives; and iii) processing time. The RSJ and PMCSJ will be compared with relation to the metrics: proportion of false and positives negatives in a "safe" R-neighboorhod of the threshold probability p. An error margin interval of size ϵ is taken from the test to avoid a judgment error caused just by the imprecision of the ground truth. The metrics are defined as $S_{FN} = \frac{\#(p-\infty,p)_{method}}{\#(p+\epsilon,p+R)_{gTruth}}$ and $S_{FP} = \frac{\#(p,\infty)_{method}}{\#(p-R,p-\epsilon)_{gTruth}}$. The execution was performed by a Intel Core i5-4200U, 1.6GHz CPU (4 threads).

4.2. Results and Discussion

Table 1 presents the metric type, S_{FP} or S_{FN}, the threshold probability p in the parenthesis, the metric value for RSJ and PMCSJ for a maximum number of 150 and 1000 simulations and a signal: positive if PMCSJ was more accurate than RSJ, negative if the winner was RSJ and neutral in case of a tie.

métrica	JEA-150	JEPMCP-150	JEPMCP-1000	sinal 1	sinal 2
$S_{FN}(0, 10)$	0,200	0,000	0,000	+	+
$S_{FN}(0, 20)$	0,000	0,056	0,000	-	
$S_{FN}(0, 50)$	0,000	0,333	0,167	-	-
$S_{FN}(0, 70)$	0,000	0,111	0,000	-	
$S_{FN}(0, 80)$	0,133	0,200	0,000	-	+
$S_{FP}(0, 20)$	0,000	0,045	0,000	-	
$S_{FP}(0, 30)$	0,000	0,077	0,000	-	
$S_{FP}(0, 40)$	0,100	0,100	0,000		+
$S_{FP}(0, 70)$	0,200	0,000	0,000	+	+
$S_{FP}(0, 90)$	0,100	0,000	0,000	+	+

Table 1. Comparing JEA and JEPMCP accuracies (just non tied scenarios).

The non parametric signal test was used to evaluate if the differences observed were statistically significant. The test evaluated the two set of hypothesis: "Set 1: (H_0) PMCSJ is so accurate as RSJ" against "(H_1) PMCSJ is less accurate than RSJ" when comparing RSJ-150 with PMCSJ-150 and "Set 2: (H_0) PMCSJ is so accurate as RSJ"

against "(H_1) PMCSJ is less accurate than RSJ" when comparing RSJ-150 with PMCSJ-1000. The test statistic **t** is given by the number of "+" considering just the non tied cases. For set 1, the p-value was 0,254 which does not raise statistically significant evidence in favor of a relevant superiotity of RSJ-150 when comparing with PMCSJ-150. For set 2, the p-value was 0,11 which is a more strong evidence in favor of a PMCSJ-1000 superiotity against RSJ-150.

RSJ took an average 257 seconds to perform the deforestation X (vegetation U wildfire) join against 64 of PMCSJ-150 and 73 of PMCSJ-1000. Thus, PMCSJ was able to be more efficient while keeping at least the same accuracy as RSJ, being a preferable method to perform a probabilistic spatil join.

5. Conclusion

The present work built and applied a variant of the Monte Carlo method and a consequence of the Chebyshev inequality to create a geneneralist probabilistic spatial join solution (PSJ) - which we call PMCSJ. The experiments showed that PMCSJ are: a) generalist concerning the positional error distribution; b) accurate; and c) efficient. Previous correlated works either were not a generalist solution for PSJ or presented a poorer performance. PMCSJ is shown to be a better generalist method for PSJ concerning the these three requirements.

The high Monte Carlo simulations cost are mitigated both by the progressive approach which applies just the required number of Monte Carlo simulations and the probabilistic filtering step which avoid several unecessary Monte Carlo simulations by applying a probabilistic and geneneralist version of the minimum bounding rectangles, which we call confidence rectangles. Both are contributions of our work.

Future works may try to build more powerful filtering steps, for example by finding a more tight shift value for confidence rectangles which still guarantees that the non intersection of two of them imply that the geometry inside them also do not intersect with a probability of at least p. Another possibility is trying to adapt the two spatial join steps to specifities of a family distribution - such as the Exponential Family, for example. That would keep the generalist requirement while advancing in performance and accuracy.

References

Arge, L., Procopiuc, O., Ramaswamy, S., Suel, T., and Vitter, J. S. (1998). Scalable sweeping-based spatial join. In *VLDB*, volume 98, pages 570–581. Citeseer.

Brinkhoff, T., Kriegel, H.-P., and Seeger, B. (1993). *Efficient processing of spatial joins using R-trees*, volume 22. ACM.

Dai, X., Yiu, M. L., Mamoulis, N., Tao, Y., and Vaitis, M. (2005). Probabilistic spatial queries on existentially uncertain data. In *International Symposium on Spatial and Temporal Databases*, pages 400–417. Springer.

Elmasri, R. (2008). *Fundamentals of database systems*. Pearson Education India.

Huang, Y.-W., Jing, N., and Rundensteiner, E. A. (1997). Spatial joins using r-trees: Breadth-first traversal with global optimizations. In *VLDB*, volume 97, pages 25–29. Citeseer.

Jacox, E. H. and Samet, H. (2003). Iterative spatial join. *ACM Transactions on Database Systems (TODS)*, 28(3):230–256.

Ljosa, V. and Singh, A. K. (2008). Top-k spatial joins of probabilistic objects. In *2008 IEEE 24th International Conference on Data Engineering*, pages 566–575. IEEE.

Lo, M.-L. and Ravishankar, C. V. (1994). Spatial joins using seeded trees. In *ACM SIGMOD Record*, volume 23, pages 209–220. ACM.

Luo, G., Naughton, J. F., and Ellmann, C. J. (2002). A non-blocking parallel spatial join algorithm. In *Data Engineering, 2002. Proceedings. 18th International Conference on*, pages 697–705. IEEE.

Mishra, P. and Eich, M. H. (1992). Join processing in relational databases. *ACM Computing Surveys (CSUR)*, 24(1):63–113.

Ni, J., Ravishankar, C. V., and Bhanu, B. (2003). Probabilistic spatial database operations. In *International Symposium on Spatial and Temporal Databases*, pages 140–158. Springer.

Openshaw, S. (1989). Learning to live with errors in spatial databases. *Accuracy of spatial databases*, pages 263–276.

Patel, J. M. and DeWitt, D. J. (1996). Partition based spatial-merge join. In *ACM SIGMOD Record*, volume 25, pages 259–270. ACM.

Patel, J. M. and DeWitt, D. J. (2000). Clone join and shadow join: two parallel spatial join algorithms. In *Proceedings of the 8th ACM international symposium on Advances in geographic information systems*, pages 54–61. ACM.

Pfoser, D. and Jensen, C. S. (1999). Capturing the uncertainty of moving-object representations. In *International Symposium on Spatial Databases*, pages 111–131. Springer.

Wolfson, O., Sistla, A. P., Chamberlain, S., and Yesha, Y. (1999). Updating and querying databases that track mobile units. In *Mobile Data Management and Applications*, pages 3–33. Springer.

Yu, X. and Mehrotra, S. (2003). Capturing uncertainty in spatial queries over imprecise data. In *International Conference on Database and Expert Systems Applications*, pages 192–201. Springer.

Zhang, R., Qi, J., Lin, D., Wang, W., and Wong, R. C.-W. (2012). A highly optimized algorithm for continuous intersection join queries over moving objects. *The VLDB JournalâThe International Journal on Very Large Data Bases*, 21(4):561–586.