Local Co-location Pattern Detection: A Summary of Results

Yan Li
Department of Computer Science & Engineering, University of Minnesota, Minneapolis, MN, USA
lixx4266@umn.edu

Shashi Shekhar
Department of Computer Science & Engineering, University of Minnesota, Minneapolis, MN, USA
shekhar@umn.edu

Abstract

Given a set of spatial objects of different features (e.g., mall, hospital) and a spatial relation (e.g., geographic proximity), the problem of local co-location pattern detection (LCPD) pairs co-location patterns and localities such that the co-location patterns tend to exist inside the paired localities. A co-location pattern is a set of spatial features, the objects of which are often related to each other. Local co-location patterns are common in many fields, such as public security, and public health. For example, assault crimes and drunk driving events co-locate near bars. The problem is computationally challenging because of the exponential number of potential co-location patterns and candidate localities. The related work applies data-unaware or clustering heuristics to partition the study area, which results in incomplete enumeration of possible localities. In this study, we formally defined the LCPD problem where the candidate locality was defined using minimum orthogonal bounding rectangles (MOBRs). Then, we proposed a Quadruplet & Grid Filter-Refine (QGFR) algorithm that leveraged an MOBR enumeration lemma, and a novel upper bound on the participation index to efficiently prune the search space. The experimental evaluation showed that the QGFR algorithm reduced the computation cost substantially. One case study using the North American Atlas-Hydrography and U.S. Major City Datasets was conducted to discover local co-location patterns which would be missed if the entire dataset was analyzed or methods proposed by the related work were applied.

2012 ACM Subject Classification Information systems → Geographic information systems, Information systems → Data mining

Keywords and phrases Co-location pattern, Participation index, Spatial heterogeneity

Digital Object Identifier 10.4230/LIPIcs.GIScience.2018.10

1 Introduction

Given instances of different spatial features (e.g., mall, hospital) and a spatial relation (e.g., geographic proximity), the problem of local co-location pattern detection (LCPD) pairs co-location patterns and localities such that the co-location patterns tend to exist inside the paired localities. A co-location pattern is a set of spatial features, the instances of which are often related to each other. The LCPD problem is one of the variants of co-location pattern detection problem, which focuses on detecting co-location patterns globally in the entire dataset [9]. Intuitively, if a co-location pattern is infrequent relative to all input instances, it may be neglected in the entire dataset, but more easily found in a subset of the dataset around its spatial footprint. The uneven distribution of spatial features in the space, i.e., spatial heterogeneity, is common, so the local existence of co-location patterns in an area is

10th International Conference on Geographic Information Science (GIScience 2018).
Editors: Stephan Winter, Amy Griffin, and Monika Sester; Article No. 10; pp. 10:1–10:15
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

not unusual. For example, high NOx emissions from buses may occur with certain engine events only around the bus depot where the route starts, since the engines have not warmed enough to perform efficiently. Other examples include high NOx emission and elevation change in rural areas as illustrated in the Volkswagen emissions scandal [8], and assault crimes and drunk driving events near bars [10]. Because of its societal importance, LCPD has attracted growing attention recently.

In this paper, we will focus on detecting local co-location patterns with the locality defined using minimum orthogonal bounding rectangles (MOBRs). An MOBR is a rectangle with sides parallel to the coordinate system. It is widely used as an approximation of complex shapes by minimally enclosing them [13]. However, the enumeration of MOBRs is computationally challenging. Given a set of spatial objects in a 2-dimensional space, the number of the set's subsets is exponentially related to its cardinality. Each of the subsets has an MOBR, so the number of MOBRs is also exponentially related to the number of the input objects. Moreover, the relationship between the participation index, a widely adopted metric for co-location patterns [9], in any pair of localities cannot be determined without considering the distribution of spatial objects within them.

The related work on the LCPD problem falls into two categories. The first line of research applies data-unaware space-partitioning heuristics (e.g. Quadtree, grid), which ignores the spatial distribution of data and may break up potential localities. The second class defines localities using clusters of spatial objects or co-location instances, but neglects other localities without a cluster.

Contributions. To detect local co-location patterns in all rectangular localities with sides parallel to the coordinate system, we first formally define the LCPD problem. Then, we present a Quadruplet & Grid Filter-Refine algorithm that leverages an MOBR enumeration lemma, and a novel upper bound on the participation index. The experimental evaluation shows that the proposed algorithm reduces the computation cost substantially. One case studies on North American Atlas-Hydrography and U.S. Major City Datasets was conducted to discover local co-location patterns which would be missed if the entire dataset was analyzed or methods proposed by the related work were applied.

This paper is organized as follows: In §2, we explain the basic concepts and formally define our local co-location pattern detection problem. §3 reviews the related literature. §4 presents our algorithms for solving the problem, whose evaluation is given in §5. §6 concludes the paper and presents our future work.

2 Basic Concepts and Problem Statement

2.1 Basic Concepts

Huang et al. define the input, output and the interest measures for detecting co-location patterns globally through data in [9].

Each spatial **object**, composed of a boolean **feature** (e.g., mall, hospital) and a spatial location, can be related to others through a spatial **relation** (e.g., neighborhood). A **co-location pattern** is a set of features. An instance of a co-location pattern is a set of objects of every distinct feature in the pattern which can form a clique given the input relation. In the dataset shown in Figure 1, there are 20 objects of feature f_A (circle) and 18 objects of feature f_B (triangle), and the related objects are linked. Only one co-location pattern, $\{f_A, f_B\}$, exists, and it has 8 instances.

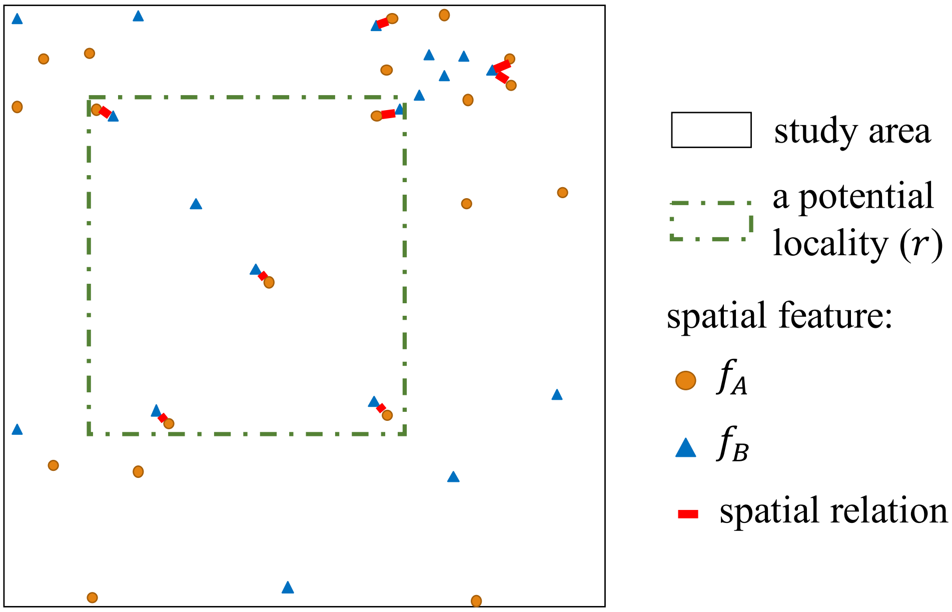


Figure 1 A local co-location pattern $< \{f_A, f_B\}, r >$.

The **participation ratio** of a feature f_i in a co-location pattern C, $pr(C, f_i)$, is the fraction of objects of the feature participating in instances of the pattern. The **participation index** of the pattern, $pi(C)$, is the minimal participation ratio of the features in the pattern. In Figure 1, for the co-location pattern $C = \{f_A, f_B\}$, $pr(C, f_A) = \frac{8}{20}$ and $pr(C, f_B) = \frac{7}{18}$, so $pi(C) = \frac{7}{18}$.

By extending these concepts, we introduce the following ones for the LCPD problem.

The **study area** is defined as the minimum orthogonal bounding rectangle (MOBR) of all input objects, whose subsets are **localities**. A **local co-location pattern** is a pair of a co-location pattern (C) and a locality (r), in the form of $< C, r >$. Its instances and interest measure are the corresponding values of its co-location pattern in its locality. A locality where objects of features in a co-location pattern tend to be related to each other (determined by a participation index threshold) is called the pattern's prevalence locality.

In Figure 1, for a local co-location pattern $C_r =< \{f_A, f_B\}, r >$, there are 5 instances, while $pr(C_r, f_A) = \frac{5}{5}$, $pr(C_r, f_B) = \frac{5}{6}$, and $pi(C_r) = \frac{5}{6}$. If the participation index threshold is 0.5, r is a prevalence locality of the pattern $\{f_A, f_B\}$.

2.2 Problem Statement

Based on the above concepts, we can formally define the LCPD problem as follows:
Input:
- A set of spatial objects.
- A spatial relation on the objects.
- A participation index threshold θ.
- A co-location instance number threshold γ.

Output: Local co-location patterns with participation index $\geq \theta$ and the number of instances $\geq \gamma$.

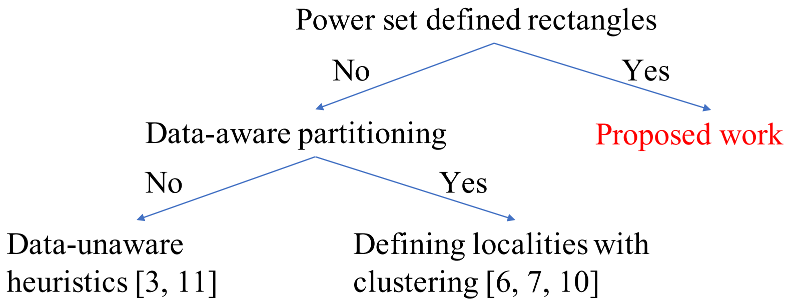

Power set defined rectangles
No
Yes
Data-aware partitioning
Proposed work
No
Yes
Data-unaware heuristics [3, 11]
Defining localities with clustering [6, 7, 10]

Figure 2 The related work.

Objective: Computational efficiency.
Constraints:

- Correctness and completeness of the result set.
- The co-location instance number threshold $\gamma \geq 2$.
- The locality of a local co-location pattern is the MOBR of its co-location instances.

If given the objects and relation in Figure 1, as well as thresholds $\theta = 0.5$ and $\gamma = 3$, $< \{f_A, f_B\}, r >$ is one of the eligible results with a participation index of $\frac{5}{6}$ and 5 instances. The co-location instance number threshold is set to prevent the problem from degradation. A locality containing only one co-location instance may be a prevalence locality, but it is meaningless.

The MOBRs of a set of co-location instances, which are the localities detected by the algorithms, can be regarded as the representatives of the infinite number of arbitrarily rectangles with sides parallel to the coordinate system according to the following lemma.

▶ **Lemma 1.** *Given any arbitrarily rectangular prevalence locality of a co-location pattern with sides parallel to the coordinate system, the MOBR of the pattern's instances within it is also a prevalence locality of the pattern.*

Proof. For any feature f in a co-location pattern C, let n_r and n_{MOBR} denote the number of objects of f in an arbitrary rectangular prevalence locality r of C and the MOBR of C's instances in r, while m_r and m_{MOBR} denote the number of those participating in C's instances. Thus, $pr(< C, r >, f) = \frac{m_r}{n_r}$, while $pr(< C, MOBR >, f) = \frac{m_{MOBR}}{n_{MOBR}}$. According to the definition of MOBR, and that $MOBR \in r$, we have $m_r = m_{MOBR}, n_r \geq n_{MOBR}$, so $\frac{m_r}{n_r} \leq \frac{m_{MOBR}}{n_{MOBR}}$. Now that $\frac{m_{MOBR}}{n_{MOBR}} \geq \frac{m_r}{n_r} \geq pi(< C, r >) \geq \theta$, the MOBR is a prevalence locality as well. ◀

3 Related Work and Limitations

In order to solve the LCPD problem, many methods have been proposed, which can be generalized into two steps. The first step is partitioning the study area into potential localities based on certain heuristics, which is followed by checking the eligibility of the localities. Based on whether the heuristics are data-aware, these methods belong to two classes (the right branch in Figure 2).

A good example using data-unaware heuristics is [3] in which Celik et al. use a QuadTree structure to divide the study area into localities, but it requires sophisticated domain knowledge to predefine localities. In another example, a grid is used to divide the study area into cells, and arbitrary subgraphs of the cells' neighbor graph are regarded as localities [12]. Both approaches share the same limitation with others using data-unaware heuristics, that is, the partitioning scheme employed is independent of the spatial distribution of the data, which may break up potential localities [10].

The other class of methods using data-aware heuristics defines localities with clusters of spatial objects or co-location instances. In [7], localities grow from initial localities with high objects concentration. Mohan et al. define localities as areas delineated by neighbor graphs of spatial objects [10]. Deng et al. explore footprints of co-location instance clusters with an adaptive density threshold as localities [6]. These methods are not complete because localities without object or co-location instance concentrations may be eligible as well.

Our proposed work, on the other hand, detects local co-location patterns in all rectangular localities with sides parallel to the coordinate system, so the method will enumerate the MOBRs determined by all subsets of co-location instances (the elements in co-location instances' power set). Consider the dataset shown in Figure 1 as an example. If the participation index threshold is set as 0.6, the co-location pattern $\{f_A, f_B\}$ is not a eligible pattern globally through the data, because its participation index is $\frac{7}{18}$. However, our proposed work will find a prevalence locality for the pattern (green dash rectangles in Figure 3a), where the participation index is $\frac{5}{6}$. Contrarily, The participation index in the locality determined by the cluster of co-location instances shown in Figure 3b is $\frac{3}{7}$, while Figure 3c and 3d present the localities with the highest possible participation index if the study area is partitioned using the Quadtree and grid in them, where the participation index is $\frac{3}{7}$ in both cases. None of the currently available results in eligible patterns, so it is obvious that the proposed work will detect more complete results than the relate work.

4 Approach

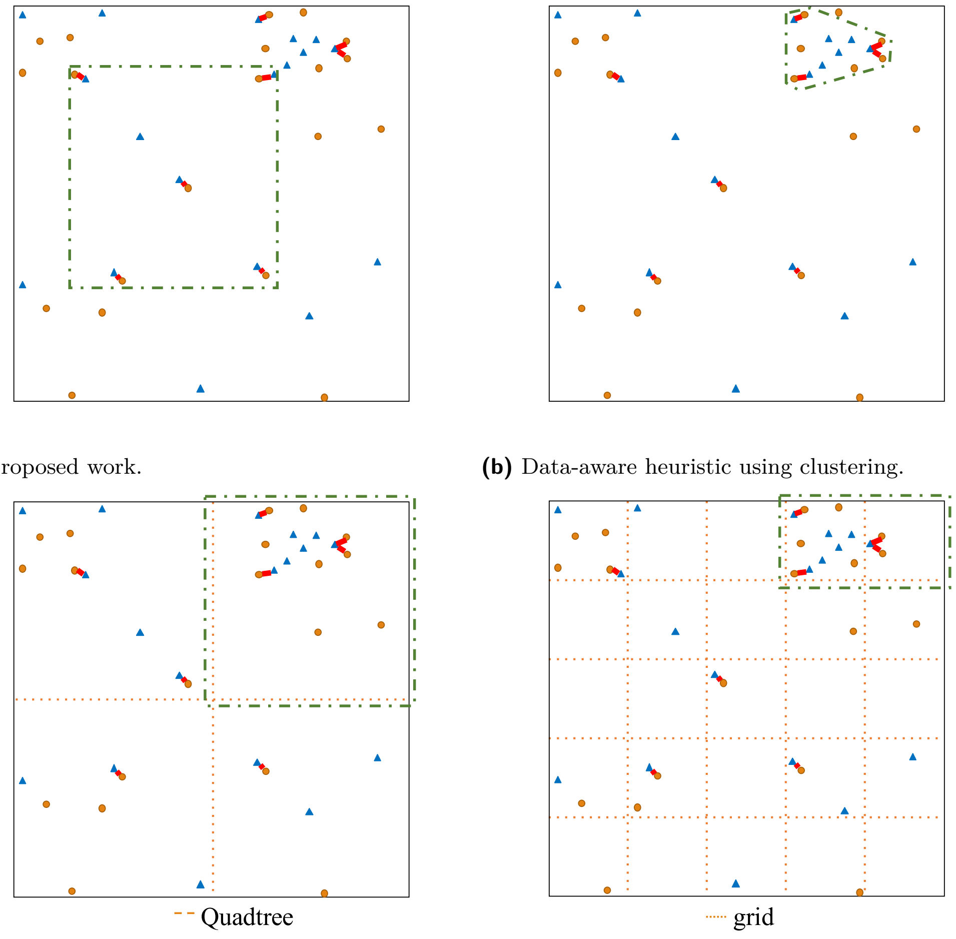
We begin this section by introducing a baseline algorithm for the LCPD problem. Then, we present two refinements: a Quadruplet (Quad) algorithm as well as a Quadruplet & Grid Filter-Refine (QGFR) algorithm, to reduce the computational cost without impairing correctness and completeness.

The pseudo-code of the general algorithm framework is shown in Algorithm 1. In this framework, all possible co-location patterns of the features associated with the input objects are enumerated in line 2-11. The instances of each co-location pattern are generated as the input of an MOBR-generating function MOBRGenerator (line 4), and the MOBRs obtained from this function are enumerated to detect the prevalance ones (line 4-10). Consider the dataset in Figure 1 as an example. In this case, F has two elements: f_A and f_B, so there is only one possible co-location pattern, $\{f_A, f_B\}$, whose 7 instances are saved in CI (line 3). The locality r is one of the MOBRs to be enumerated. There are 5 instances within it, and the participation index is $\frac{5}{6}$. Both metrics will be compared with the thresholds to determine whether $< \{f_A, f_B\}, r >$ is an eligible result.

In this study, we focus on reducing the number of MOBRs enumerated for each co-location pattern (i.e., improving function MOBRGenerator$(\cdot)$), but adopt Apriori-like algorithms to reduce the number of possible co-location patterns [9, 6], and the state-of-the-art algorithms to generate co-location instances [9, 14].

4.1 Baseline Algorithm

As already mentioned, we focus on localities defined as the MOBRs of subsets of co-location instances. In the function MOBRGenerator$(\cdot)$ of the baseline algorithm, we will enumerate all arbitrary subsets of the input co-location instances, and generate an MOBR for each of



(a) Proposed work.

(b) Data-aware heuristic using clustering.

(c) Data-unaware heuristic using Quadtree.

(d) Data-unaware heuristic using a grid.

Figure 3 Comparison between related work. (Better in color.)

them. If each co-location pattern has n_{ci} instances on average, there will be $2^{n_{ci}}$ subsets, resulting in $2^{n_{ci}}$ MOBRs. Thus, the computational complexity of this baseline algorithm is $O(k2^{n_{ci}})$, where k is the number of possible co-location patterns.

4.2 Quad-Element Algorithm

Our first improvement is based on an MOBR enumeration lemma:

▶ **Lemma 2.** *Given a set s of n points in a two-dimensional plane, the set of MOBRs for arbitrary subsets of s is the same as the set of MOBRs for arbitrary subsets with cardinality ≤ 4 of s.*

Proof. Assume that there exists an MOBR for a subset (sub) with cardinality > 4 that is not an MOBR for a subset with cardinality ≤ 4.

Let $x_{min}, x_{max}, y_{min}, y_{max}$ denote the minimum and maximum of the x, y coordinates of the points in sub. There must exist points a, b, c, and d (which may be the same) in sub such that $x_a = x_{min}, x_b = x_{max}, y_c = y_{min}, y_d = y_{max}$. Thus, the MOBR for sub is the same as that for $\{a, b, c, d\}$, which is a subset of s with cardinality ≤ 4, resulting in a contradiction with the assumption. ◀

Algorithm 1 General algorithm framework.

```
Require:
    $Obj$: A set of objects;
    $R$: A spatial relation over objects in $Obj$;
    $\theta$: Participation index threshold;
    $\gamma$: Co-location instance number threshold.
Ensure: Local co-location patterns with participation index $\geq \theta$ and the number of instances $\geq \gamma$.
$F \leftarrow$ all spatial features in $Obj$;
for all possible patterns $C$ of $F$ do
    $CI \leftarrow$ co-location instances of $C$;
    for all $mobr \in$ MOBRGenerator($CI$) do
        $p \leftarrow$ the participation index of $C$ in $mobr$;
        $n \leftarrow$ the number of $C$'s instances in $mobr$;
        if $p \geq \theta$ and $n \geq \gamma$ then
            Add $< cp, mobr >$ to the result.
        end if
    end for
end for
```

Lemma 2 indicates that the enumeration cost of a co-location pattern's MOBRs can be reduced from 2^n to n^4 without affecting completeness. By changing the function MOBRGenerator($\cdot$) to generate the MOBRs of subsets with cardinality ≤ 4 of CI we can get the Quadruplet (Quad) algorithm with computational complexity of $O(kn_{ci}^4)$.

4.3 Quadruplet & Grid Filter-Refine Algorithm

Our definition of localities determines that a small displacement of any co-location instance that defines a locality's boundary will create a new locality, so there are lots of localities overlapping each other. If we can classify them into groups according to the areas they share, and apply a filter on each group instead of on individuals, the number of localities to be enumerated can be reduced further. Based on this idea, we proposed the second improvement: the Quadruplet & Grid Filter-Refine (QGFR) Algorithm.

The pseudo-code of the function MOBRGenerator($\cdot$) in the QGFR algorithm is shown in Algorithm 2. Because a grid-based filter is applied, three new parameters are added, namely, a threshold of the participation index, a threshold of the number of co-location instances, and the cell size of the grid covering the entire study area. The first step of the function is saving the active cells of the input co-location pattern C (i.e., the cells overlapping C's instances) in AC (line 2). A cell overlapping a co-location instance means that the intersection of the cell and the MOBR of this instance is nonempty. For example, Cells 1, 2, and 3 in Figure 4 are active cells of the pattern $\{f_A, f_B\}$. After getting the active cells, we will use their MOBRs (cMOBR) as an approximation of the MOBRs of C's instances (iMOBR). The cells in a cMOBR are classified into two parts. The cells adjacent to the cMOBR's boundary are named as *bounding* cells, while the others are the *bounded* cells. In Figure 4, a cMOBR is delineated by a red solid rectangle, while its bounding and bounded cells are filled with a hash pattern and a solid color respectively. The boundary of each iMOBR has the following property:

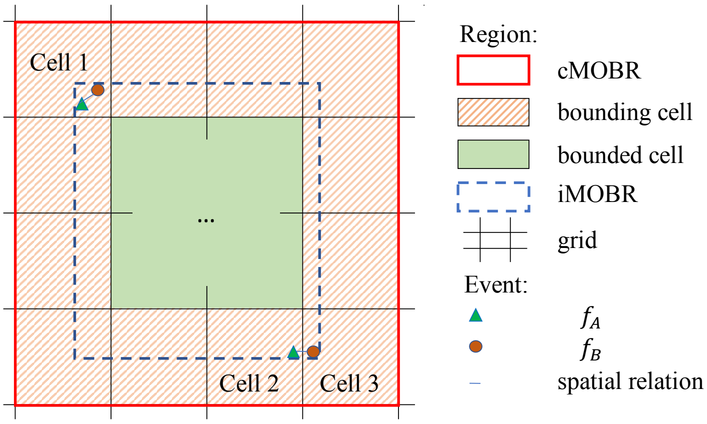


Figure 4 Grid cells and MOBRs (better in color).

▶ **Lemma 3.** *The boundary of any iMOBR must be within the bounding cells of one and only one cMOBR.*

The proof of this lemma is straightforward. If the boundary of an iMOBR is not within the bounding cells of a cMOBR, at least one of its four edges does not pass active cells, which is impossible. If two cMOBRs share the same bounding cells containing an iMOBR's boundary, they must be the same. Therefore, we define that an iMOBR is in a cMOBR if its boundary is within the bounding cells of the cMOBR. For example, an iMOBR delineated by a dash rectangle in Figure 4 is in the plotted cMOBR. Because each iMOBR is in a unique cMOBR, by enumerating the iMOBRs in each cMOBR, we can enumerate all iMOBRs just once. In the pseudo-code, we enumerate all cMOBRs using Lemma 2 (line 3-10).

To eliminate the cMOBRs in which no iMOBR is eligible, we introduce an upper bound (MaxPI bound), $\eta(< C, \text{cMOBR} >)$, for the participation index of a local co-location pattern composed of a co-location pattern C and any iMOBR in a cMOBR of C. The MaxPI bound is based on an upper bound for the participation ratio, which can be stated as:

▶ **Lemma 4.** *The upper bound, $\zeta(< C, \text{cMOBR} >, f)$, for the participation ratio of a feature f in a local co-location pattern composed of a pattern C and any iMOBR in a cMOBR of C is*

$$\zeta(< C, cMOBR >, f) = \frac{po(C, f, \text{cMOBR})}{o(f, \text{bounded}) + po(C, f, \text{bounding})}$$

$\forall$ iMOBR in cMOBR.

Table 1 describes the notation used in the above formula.

Table 1 Symbols used in Lemma 4.

Number of objects of f in a locality r		
Participating in C	Not participating in C	All
$po(C, f, r)$	$npo(C, f, r)$	$o(f, r)$

where r can take values of "all cells" (cMOBR), "bounding cells" (bounding), or "bounded cells" (bounded) of the cMOBR, or the "actual iMOBR" (iMOBR), or the "intersection of iMOBR and bounding cells" (extra). The proof is as follows:

Proof.

$$pr(<C, iMOBR>, f) = \frac{po(f, C, \text{iMOBR})}{o(f, \text{iMOBR})} = \frac{po(f, C, \text{bounded}) + po(f, C, \text{extra})}{o(f, \text{bounded}) + o(f, \text{extra})}$$
$$= \frac{po(f, C, \text{bounded}) + po(f, C, \text{extra})}{o(f, \text{bounded}) + po(f, C, \text{extra}) + npo(f, C, \text{extra})}.$$

Because $npo(f, C, \text{extra}) \geq 0$,

$$pr(<C, iMOBR>, f) \leq \frac{po(f, C, \text{bounded}) + po(f, C, \text{extra})}{o(f, \text{bounded}) + po(f, C, \text{extra})}.$$

Because extra $\in$ bounding, $0 \leq po(f, C, \text{extra}) \leq po(f, C, \text{bounding})$. Meanwhile, $\frac{po(f,C,\text{bounded})}{o(f,\text{bounded})} \leq 1$. Thus,

$$pr(<C, iMOBR>, f) \leq \frac{po(f, C, \text{bounded}) + po(f, C, \text{bounding})}{o(f, \text{bounded}) + po(f, C, \text{bounding})}$$
$$= \frac{po(f, C, \text{cMOBR})}{o(f, \text{bounded}) + po(f, C, \text{bounding})}.$$

◀

Based on the definition of the participation index, we can define the MaxPI bound as the smallest upper bound of the participation ratio of any feature in the local co-location pattern, i.e.,

$$\eta(<C, \text{cMOBR}>) = min_{f_i \in C}(\zeta(<C, \text{cMOBR}>, f_i)).$$

Given a participation index threshold θ, if $\eta(<C, \text{cMOBR}>) < \theta$, there will not be any eligible iMOBR in this cMOBR. In the pseudo-code, the MaxPI bound of C in every one of its cMOBRs, together with the number of instances, is compared with the thresholds to determine whether enumerating the iMOBRs in the current cMOBR is necessary.

Algorithm 2 Function MOBRGenerator in QGFR algorithm.

```
Require:
    $CI$: A set of instances of a co-location pattern $C$;
    $\theta$: Participation index threshold;
    $\gamma$: Co-location instance number threshold;
    $l$: The size of each grid cell.
Ensure: MOBRs of $CI$'s subsets.
function MOBRGENERATOR($CI, \theta, \gamma, l$)
    $AC \leftarrow$ active cells of $C$;
    for all $subAC$(with cardinality $\leq 4$) $\subseteq AC$ do
        $cmobr \leftarrow$ the MOBR of $subAC$;
        $\eta \leftarrow$ MAXPI($C, cmobr$);
        $n \leftarrow$ the number of $C$'s instances in $cmobr$;
        if $\eta \geq \theta$ and $n \geq \gamma$ then
            Add iMOBRs in $cmobr$ to the result.
        end if
    end for
end function
```

Assuming that each co-location pattern has n_{ac} active cells on average, and the number of iMOBRs in each cMOBR is q, the computational complexity is $O(kn_{ac}^4 q)$. If q can be

Table 2 Parameters for the experiments.

Symbol	Meaning
n_{cp}	Number of core co-location patterns
n_{cc}	Core co-location patterns' cardinality
n_{ci}	Number of instances of each pattern
n_i	Number of input objects
n_f	Number of input features
Grid size	Cell's edge length of the grid used in the QGRF algorithm

treated as a constant, because n_{ac} is much less than n_{ci} in most cases, the computational cost of the QGFR algorithm is much lower than that of the Quad. Because we have proved that in this algorithm all MOBRs of co-location instances are evaluated once and only eligible results are returned, we maintain the correctness and completeness of the algorithm through the performance improvement.

5 Experimental Evaluation and Case Studies

In this section, we evaluate the baseline, Quad, and QGFR algorithm using synthetic data and a Chicago crime dataset [4], followed by one case study using the North American Atlas - Hydrography dataset from the U.S. Geological Survey [11] and the dataset of the U.S. major cities from Esri.

5.1 Experiments

The goal of the experiments was twofold: (a) evaluate the effect of the performance refinements of the proposed Quad algorithm and QGFR algorithm compared with the baseline algorithm. (b) determine the robustness of the QGFR algorithm given different inputs.

According to our analysis in §4, the computational complexity of the three algorithms are $O(k2^{n_{ci}}), O(kn_{ci}^4)$, and $O(kn_{ac}^4 q)$ respectively, where n_{ci} is the number of co-location instances per pattern, n_{ac} is the number of active cells per pattern, k is the number of co-location patterns, and q is the average number of iMOBR in each cMOBR. To evaluate the performance refinements, we studied the following two questions: (1) What is the effect of the number of co-location instances? (2) What is the effect of the number of co-location patterns? To determine the robustness, we asked how well the QGFR algorithm performed under different size of grid cells.

To answer these questions, we designed experiments as shown in Figure 5. The synthetic and the real-world data (a Chicago crime dataset) were generated with controlled parameters. In the simulation, three algorithms were executed with the grid cell size as a parameter. The performance was evaluated and compared using the run time of each algorithm. The platform for the simulation was Microsoft .NET Framework 4.5 on a computer with Intel(R) Core(TM) i7-4770 3.40 GHz CPU and 32 GB RAM. The parameters in the experiments are shown in Table 2.

5.1.1 Synthetic data generation

A point distribution with co-location patterns is often modeled as an aggregated point process [9, 2, 6]. Commonly used point processes include the Poisson cluster process [1] and Matérn's cluster process [5]. In order to ensure the existence of local co-location patterns, we made two changes on the steps used in [2], including:

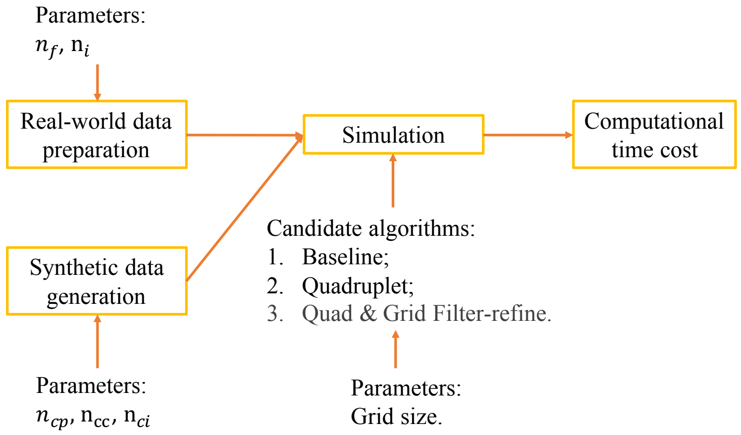


Figure 5 Experiment design.

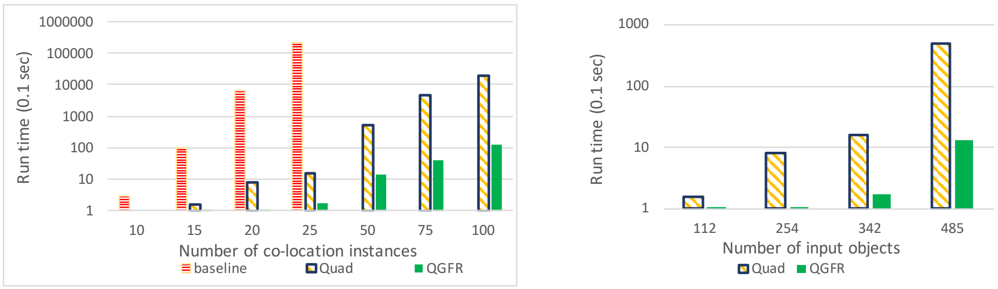
- Randomly select a rectangular region in the study area as a prevalence locality for each co-location pattern.
- In each co-location pattern's prevalence locality, ensure that at least 4 instances of the pattern are generated, and that no noise object of the features in the pattern is generated.

Because the subsets of a co-location pattern are also co-location patterns, when generating the synthetic data, we named the patterns which were not subsets of other patterns core patterns. The study area size was set to 10000 × 10000. The spatial relation was a neighborhood with a radius of 10. The number of noise objects of each feature was set to $4 \times n_{ci}$.

5.1.2 Experimental results

Effect of the number of co-location instances. The experiments were conducted with both synthetic and real-world data. The synthetic data was generated by fixing $n_{cp} = 2$ and $n_{cc} = 3$, but changing n_{ci}, whose results were shown in Figure 6a. The computational cost of the baseline algorithm, as expected, increased exponentially with n_{ci}, and was much larger than that of the two proposed algorithms, so its run time was not included when $n_{ci} = 50, 75$, or 100. The run time of the Quad algorithm was much longer than that of the QGFR algorithm, and it also increased faster than the latter with increasing n_{ci}. The experiment with the Chicago crime dataset was conducted by fixing $n_f = 3$ but varying n_i. By increasing the number of input objects in a fixed study area, we increased the number co-location instances indirectly. The results (Figure 6b) also shown that the advantage of the QGFR algorithm increased as the number of input objects grew.

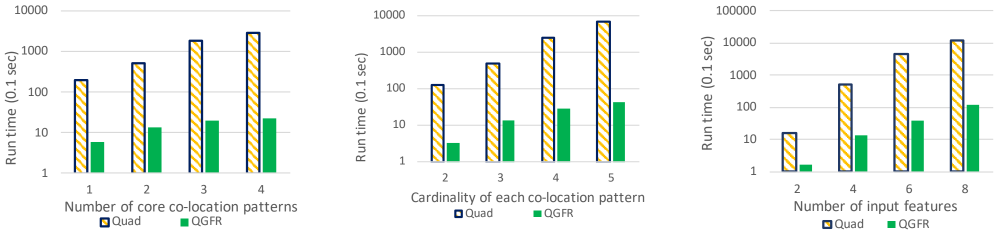
Effect of the number of co-location patterns. Since the number of co-location patterns is determined by both the number of core co-location patterns and their cardinalities, we conducted two controlled experiments with synthetic data and one with the Chicago crime dataset on them. Figure 7a and Figure 7b presented the results of experiments with the synthetic data. In Figure 7a $n_{cc} = 3$ and $n_{ci} = 50$ but n_{cp} changed, while in Figure 7b $n_{cp} = 2$ and $n_{ci} = 50$ but n_{cc} changed. Figure 7c shown the results using the real-world data, where the number co-location pattern was increased by increasing the number of input features. In all the cases, the growing number of co-location patterns increased the advantage of the QGFR algorithm over the Quad algorithm.



(a) Results with synthetic data.

(b) Results with the Chicago crime dataset.

Figure 6 Effect of the number of co-location instances.



(a) Effect of the number of core co-location patterns.

(b) Effect of each co-location pattern's cardinality.

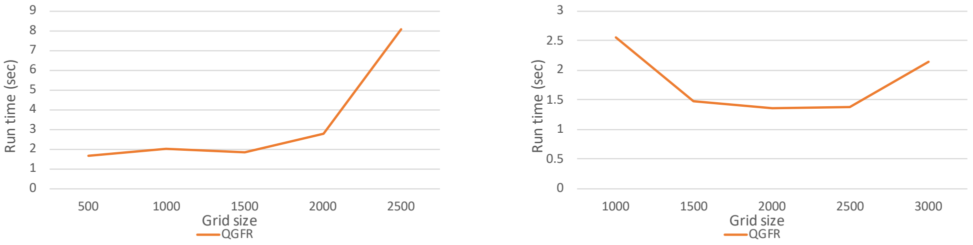
(c) Effect of the number of input features.

Figure 7 Effect of the number of co-location patterns

Effect of the size of grid cells. The sensitivity analysis was done through two controlled experiments where the same synthetic and real-world data but different grid cell size were used. The parameters for the synthetic data were $n_{cp} = 2, n_{cc} = 3, n_{ci} = 50$ and those for real-world data were $n_i = 485, n_f = 4$. According to the results shown in Figure 8, the QGFR algorithm was robust with changes in the grid cell size, since the fluctuation of its run time was small when the grid cell size changed. When the grid cell size was small, the number of active cells was not much smaller than the number of co-location instances, so the performance would be improved if a larger cell size was used. As the grid cell size increased, more iMOBRs resided in a single grid cell, so the performance improvement brought about by the MaxPI bound was weakened.

5.2 Case Study using North American Atlas-Hydrography and U.S. Major City Datasets

We conducted a case study using the North American Atlas - Hydrography dataset from the U.S. Geological Survey and the data of the U.S. major cities from Esri. Other inputs included a spatial relation specified by a neighborhood radius of 50 kilometers, a participation index threshold $\theta = 0.6$, and a instance number threshold $\gamma = 20$. There were 2610 cities which represent cities in the U.S. with population of more than 10 thousand in the dataset. The number of lakes was 394. The participation index of the co-location pattern $\{city, lake\}$ was 0.33, which meant major cities were not globally co-located with lakes in the U.S. However, our proposed QGFR algorithm detected some prevalence localities, two of which were shown in Figure 9 with the zoom-in maps. In the east locality, there were 163 cities, 109 of which were co-located with lakes, while 39 out of 41 lakes were near cities, so the participation index was about 0.67. This locality could be detected by the related work as well, because if



(a) Results with synthetic data.

(b) Results with the Chicago crime dataset.

Figure 8 Effect of the size of grid cells.

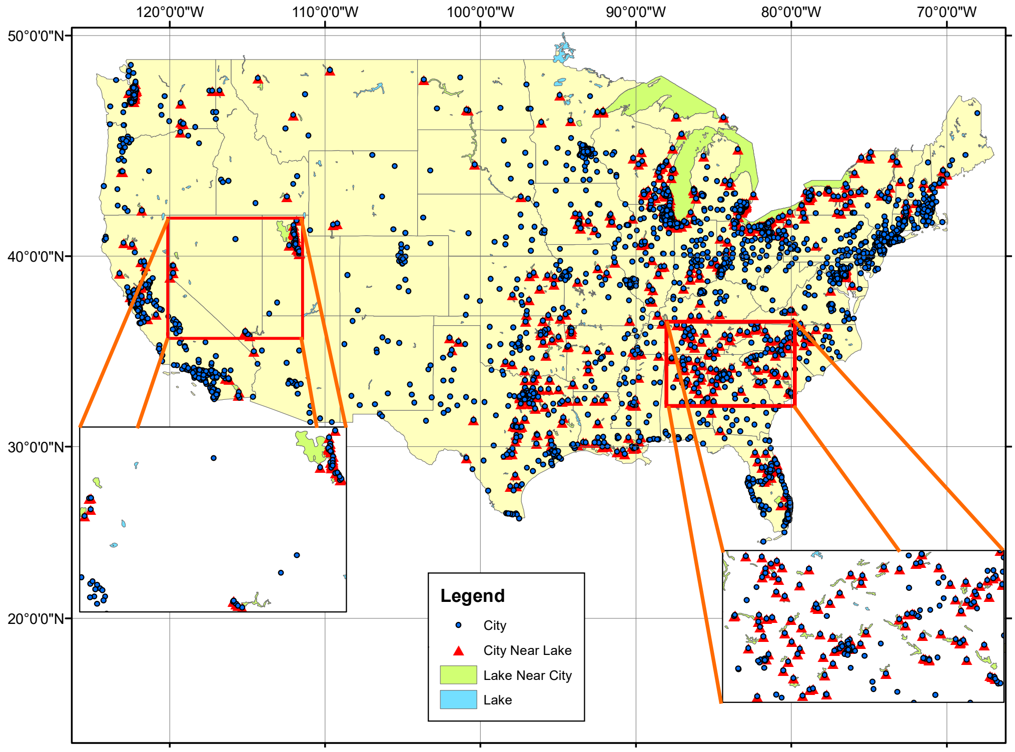


Figure 9 Case study with the hydrography and city data. Two prevalence localities of co-location pattern $\{city, lake\}$ are delineated by rectangles and shown in the zoom-in maps. (Better with color.)

we defined the density as the number of instances of a feature in a unit area, the density of both input objects and co-location instances was high (the ratio between the density of the co-location instances in the locality and that in the whole country was about 4.22). Contrarily, in the west locality, there were 35 out of 50 cities co-located with 7 out of 11 lakes, resulting in the participation index as about 0.63. In this locality, the density of the input objects and co-location instances was almost the same as that in the whole country (the ratio between the density of the co-location instances in the locality and that in the whole country was about 1.03), which meant that the locality could not be identified by the related work using clustering to define localities. The findings indicated that the co-location pattern of major cities and lakes existed not only in the southeast of the U.S where lakes concentrated but also in the west where it was drier and lakes were more valuable of the cities.

6 Conclusion and Future Work

In this paper, we formally defined the local co-location pattern detection problem, and proposed two algorithms that can efficiently solve it. The effectiveness and efficiency of the algorithms were proved theoretically and validated experimentally on synthetic and real datasets. In addition, we presented the results of one case study using the North American Atlas-Hydrography and U.S. Major City Datasets.

During the study, we noticed that the distribution of spatial events (e.g., the auto-correlation between events of the same feature) may affect the results. Our future research will take this into consideration. In addition, the distribution of events related to humans may be strongly affected by road networks especially in urban areas. Defining regions as subsets of road networks may result in richer and more meaningful results. We plan to explore this idea in our future work.

References

1 Adrian Baddeley. Spatial Point Processes and their Applications. In *Stochastic Geometry*, volume 1892 of *Lecture Notes in Mathematics*, pages 1–75. Springer, Berlin, Heidelberg, 2007. `doi:10.1007/978-3-540-38175-4\_1`.

2 S. Barua and J. Sander. Mining Statistically Significant Co-location and Segregation Patterns. *IEEE Transactions on Knowledge and Data Engineering*, 26(5):1185–1199, 2014. `doi:10.1109/TKDE.2013.88`.

3 Mete Celik, James M. Kang, and Shashi Shekhar. Zonal co-location pattern discovery with dynamic parameters. In *Data Mining, 2007. ICDM 2007. Seventh IEEE International Conference on*, pages 433–438. IEEE, 2007. URL: `http://ieeexplore.ieee.org/abstract/document/4470269/`.

4 Chicago Police Department. Crimes - 2001 to present, 2017. [Online; accessed 30-September-2017]. URL: `https://data.cityofchicago.org/Public-Safety/Crimes-2001-to-present/ijzp-q8t2`.

5 Sung Nok Chiu, Dietrich Stoyan, Wilfrid S. Kendall, and Joseph Mecke. *Stochastic Geometry and Its Applications*. John Wiley & Sons, 2013.

6 Min Deng, Jiannan Cai, Qiliang Liu, Zhanjun He, and Jianbo Tang. Multi-level method for discovery of regional co-location patterns. *International Journal of Geographical Information Science*, 31(9):1846–1870, 2017. `doi:10.1080/13658816.2017.1334890`.

7 Christoph F. Eick, Rachana Parmar, Wei Ding, Tomasz F. Stepinski, and Jean-Philippe Nicot. Finding Regional Co-location Patterns for Sets of Continuous Variables in Spatial Datasets. In *Proceedings of the 16th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, GIS '08, pages 30:1–30:10, New York, NY, USA, 2008. ACM. `doi:10.1145/1463434.1463472`.

8 Guilbert Gates, Jack Ewing, Karl Russell, and Derek Watkins. How Volkswagen's 'Defeat Devices' Worked. *The New York Times*, 2015. URL: `https://www.nytimes.com/interactive/2015/business/international/vw-diesel-emissions-scandal-explained.html`.

9 Yan Huang, Shashi Shekhar, and Hui Xiong. Discovering colocation patterns from spatial data sets: a general approach. *IEEE Transactions on Knowledge and Data Engineer-*

ing, 16(12):1472–1485, 2004. URL: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1350759.

10 Pradeep Mohan, Shashi Shekhar, James A. Shine, James P. Rogers, Zhe Jiang, and Nicole Wayant. A Neighborhood Graph Based Approach to Regional Co-location Pattern Discovery: A Summary of Results. In *Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, GIS '11, pages 122–132, New York, NY, USA, 2011. ACM. doi:10.1145/2093973.2093991.

11 USGS. North america rivers and lakes, 2018. [Online; accessed 13-February-2018]. URL: https://www.sciencebase.gov/catalog/item/4fb55df0e4b04cb937751e02.

12 Song Wang, Yan Huang, and Xiaoyang Sean Wang. Regional Co-locations of Arbitrary Shapes. In *Advances in Spatial and Temporal Databases*, pages 19–37. Springer Berlin Heidelberg, 2013. doi:10.1007/978-3-642-40235-7_2.

13 Jordan Wood. Minimum Bounding Rectangle. In Shashi Shekhar, Hui Xiong, and Xun Zhou, editors, *Encyclopedia of GIS*, pages 1232–1233. Springer International Publishing, 2 edition, 2017. doi:10.1007/978-3-319-17885-1_783.

14 Jin Soung Yoo and S. Shekhar. A Joinless Approach for Mining Spatial Colocation Patterns. *IEEE Transactions on Knowledge and Data Engineering*, 18(10):1323–1337, oct 2006. doi:10.1109/TKDE.2006.150.