Spatio-sequential patterns mining: Beyond the boundaries

Hugo Alatrista-Salas, Sandra Bringay, Frédéric Flouvat, Nazha Selmaoui-Folcher, Maguelonne Teisseire

To cite this version:

Hugo Alatrista-Salas, Sandra Bringay, Frédéric Flouvat, Nazha Selmaoui-Folcher, Maguelonne Teisseire. Spatio-sequential patterns mining: Beyond the boundaries. Intelligent Data Analysis, 2016, 20 (2), pp.293-316. 10.3233/ida-160806 . lirmm-01348460

HAL Id: lirmm-01348460
https://hal-lirmm.ccsd.cnrs.fr/lirmm-01348460

Submitted on 16 May 2020

Intelligent Data Analysis 20 (2016) 293–316
DOI 10.3233/IDA-160806
IOS Press

Spatio-sequential patterns mining: Beyond the boundaries

Hugo Alatrista-Salas[a,b,d,*], Sandra Bringay[c], Frédéric Flouvat[b], Nazha Selmaoui-Folcher[b] and Maguelonne Teisseire[a]

[a]*Irstea-TETIS, Montpellier, France*
[b]*PPME, Noumea, New Caledonia*
[c]*LIRMM, Montpellier, France*
[d]*Pontificia Universidad Católica del Perú, San Miguel, Lima, Perú*

Abstract. Data mining methods extract knowledge from huge amounts of data. Recently with the explosion of mobile technologies, a new type of data appeared. The resulting databases can be described as spatiotemporal data in which spatial information (e.g., the location of an event) and temporal information (e.g., the date of the event) are included. In this article, we focus on spatiotemporal patterns extraction from this kind of databases. These patterns can be considered as sequences representing changes of events localized in areas and its near surrounding over time. Two algorithms are proposed to tackle this problem: the first one uses *a priori* strategy and the second one is based on pattern-growth approach. We have applied our generic method on two different real datasets related to: 1) pollution of rivers in France; and 2) monitoring of dengue epidemics in New Caledonia. Additionally, experiments on synthetic data have been conducted to measure the performance of the proposed algorithms.

Keywords: Sequential patterns, spatiotemporal data mining, health risk management

1. Introduction

New mobile technologies enable information to be linked with temporal and spatial characteristics. Data mining methods must be able to fulfill the new needs generated by spatiotemporal data. For instance, the study of phenomena like dengue fever epidemics or watercourses pollution become critical due to their complex dynamic.[1] In the case of dengue epidemics, public health experts know that the evolution of the disease depends on environmental factors (e.g., climate, areas with water points, mangroves, etc.) and on interactions between human and vector transmission (e.g., the mosquito that carries the disease). However, the impact of environmental factors and their interactions remain unclear.

In this context, methods for mining spatiotemporal data provide very relevant solutions through identification without *a priori* hypothesis of relationships between variables and events characterized in time and space. For example, in our context, we will discover combinations of changes in environmental factors that lead to epidemic peaks in specific spatial configurations. To extract this kind of information,

[*] Corresponding author: Hugo Alatrista-Salas, Irstea-TETIS, 500, rue J.F. Breton 34093, Montpellier, France. E-mail: hugo.alatrista-salas@teledetection.fr.

[1] Yuang [26] describes this concept of dynamics as a *set of dynamic forces impacting the behavior of a system and its components, individually and collectively.*

we define a new type of spatiotemporal pattern called *Spatio-Sequential Patterns* or simply *S2P*, which allows us to discover the temporal evolution of events appearing not only in a studied area but also in its near surrounding. This new type of patterns have been defined in previous work [3]. An example of pattern in the dengue context is: *frequently over the past 10 years, if it rains in an area and if there are standing water and high temperatures in the neighborhood, then there is an increasing number of mosquitoes in adjacent areas, followed by an increase of dengue cases*. Such patterns are very interesting because they enable to capture the evolution of areas considering their events and events in adjacent zones.

In contrast to previous work [3], we propose in this paper, two generic algorithms to extract the S2P. The first one uses the *a priori* strategy and the second one is based on pattern-growth approach with efficient successive projections over the database. Also, the experimentations have been conducted on synthetic data and on two real spatiotemporal databases. In addition, as our approach generates a lot of patterns which are not easy to interpret by the experts, we propose in this paper, an interestingness measure to overcome this problem.

This manuscript is organized as follows: in Section 2, we review existing spatiotemporal data mining methods and we show that these methods are not suitable for our problem. Next, in Section 3, we detail the theoretical framework around the S2P. Then, we define two propositions of pruning measures in Section 4. Afterwards, in Section 5, we propose two algorithms to extract the S2P. Toward the end of this article, in Section 6, we present experiments on real and synthetic datasets. The paper ends with our conclusions and future perspectives.

2. Related works

In this section, we only focus on methods analyzing the evolution and the interaction of objects or events characteristics through space and time and we are not concerned by the trajectory issues addressed [8,14,21]. Early work addressed the spatial and temporal dimensions separately. For example, Han et al. [9] or Shekhar et al. [20] looked for spatial patterns or co-locations, i.e., subsets of features (object-types) with instances often identified as close in space. In our context, an example of co-location is: *within a radius of 200 m, mosquitoes nests are frequently found near ponds.* On the contrary, other authors as Pei et al. [19] or Liu [15] have studied temporal sequences which only take into account the temporal dimension. Tsoukatos et al. [23] have extended these works to represent sets of environmental features evolving in time. They extract sequences of characteristics that appear frequently in areas, but without taking into account the spatial neighborhood. An example of pattern obtained is: *in many areas, heavy rain occurs before the formation of a pond, followed by the development of mosquito nest*. If these two types of methods, only spatial or temporal, can be very relevant for epidemiological surveillance, they do not capture relations such as: *often, a heavy rain occurs before the formation of a pond followed in a close area by the development of mosquito nests*. In [24], Wang et al. focus on the extraction of sequences representing the propagation of spatiotemporal events in predefined time windows w.r.t. a reference location. They introduce two concepts: *Flow patterns* and *Generalized Spatiotemporal Patterns* in order to extract precisely the sequence of events that occurs frequently in some locations. Thus, the authors will be able to identify patterns like: *dengue cases appear frequently in area Z1 after the occurrence of high temperatures and the presence of ponds in area Z2*.

However, Huang et al. [11] found that all the patterns discovered with other approaches are not always relevant because they may not be statistically significant and in particular not *dense* in space and time. They therefore proposed an interestingness measure taking into account the spatial and temporal aspects

Table 1
Summarization of related work

Reference	Pattern type	Spatiotemporal properties	Interesting measure
[4,8,14]	Trajectory	Spatiotemporal sequence	Support
[20]	Set of features	Spatial co-location	Conditional probability
[5]	Set of features	Spatial co-location	Temporal and spatial prevalence
[19]	Set of features	Temporal event	Support
[23]	Set of features	Spatiotemporal event	Support
[24]	Sequences of features	Spatiotemporal spread of objects	Temporal and spatial support
[11]	Sequences of features	Spatiotemporal spread of objects	Sequence index
[17]	Set of features	Spatiotemporal event	Cascade participation index

to extract global sequence of features. However, they study events one after another. They do not take into account interactions such as: *often heavy rain and the occurrence of ponds are presented before the development of mosquito nests*. Celik et al. [5], proposed the concept of *Mixed-Drove Spatiotemporal Co-occurrence Patterns*, i.e., subsets of two or more different event-types whose instances are often located in spatial and temporal proximity (e.g., an event-type is *heavy rain* and an instance is *heavy rain in zone Z1 the 10/17/2013*). For similar reasons than Huang, they have proposed a specific monotonic composite interestingness measure based on spatial and temporal prevalence measures. However, they do not extract the frequent evolutions of even-types over time (events of each instance occur necessarily in the same time slot). For example, we can only extract patterns such as: *heavy rain, ponds and development of mosquito nests are frequently found together in lots of time slots*.

On the other hand, Mohan et al. [17] have proposed a new spatiotemporal pattern called Cascading Spatiotemporal Pattern (CSTP) that represents a partial ordered subset of event-types whose instances are located together and occur serially. In our context, an example of CSTP is: *often, after heavy rain, a mosquitoes nest and some cases of dengue are present in a zone*. This kind of patterns, which are partial ordered – as opposed to sequential patterns that are fully ordered – does not capture the interactions between a subset of features occurring in a particular area with others around. Finally, approaches proposed by Wang, Huang, Celik and Mohan can not capture the evolution of areas with regard to their set of event-types and the sets of event-types of their neighbors.

Table 1 presents a resume of related work considering the pattern type, the spatiotemporal properties and the interestingness measure.

In this paper, we describe a method for extracting spatiotemporal sequences of patterns (i.e., sequences of spatial sets of events) called Spatio-Sequential Patterns (S2P). We aim at identifying relationships such as: *the presence of dengue cases in an area is often preceded by high temperatures and the presence of water tanks in a neighboring area*. Thus, we will deal with the developments and interactions between the study area and its immediate environment. Moreover, as this kind of patterns are very difficult to mine, because of the huge generated search space, we will introduce an interestingness measure to make our approach more scalable.

3. S2P (Spatio-Sequential patterns): Concepts and definitions

In this section, we present the basic concepts around the spatio-sequential patterns.

3.1. Preliminaries

A spatiotemporal database is a structured set of information including geographic components (e.g., neighborhoods, rivers, etc.), temporal components (dates) and events (e.g., rain, wind). Such a database

Table 2
Weather changes in three zones: Z_1, Z_2 and Z_3 on December 22, 23, 24, 2013

Zone	Date	Temp	Prec	Wind	Gusts
Z_1	12/22/13	T_l	P_m	W_m	–
Z_1	12/23/13	T_m	P_m	W_l	–
Z_1	12/24/13	T_l	P_m	W_m	55
Z_2	12/22/13	T_m	P_m	W_m	–
Z_2	12/23/13	T_l	P_m	W_l	–
Z_2	12/24/13	T_l	P_l	W_m	–
Z_3	12/22/13	T_l	P_m	W_s	75
Z_3	12/23/13	T_m	P_s	W_l	–
Z_3	12/24/13	T_m	P_s	W_s	55

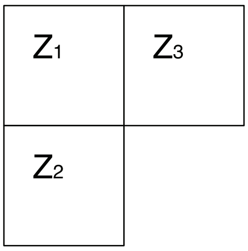


Fig. 1. Neighboring zones.

is defined as a triplet $DB = (D_{\mathrm{T}}, D_{\mathrm{S}}, D_{\mathrm{A}})$ where D_{T} is the temporal dimension, D_{S} the spatial dimension and $D_{\mathrm{A}} = \{D_{\mathrm{A}_1}, D_{\mathrm{A}_2}, \ldots, D_{\mathrm{A}_p}\}$ a set of analysis dimensions associated with attributes. The *temporal dimension* is associated with a domain of values denoted by $dom(D_{\mathrm{T}}) = \{T_1, T_2, \ldots, T_t\}$ where $\forall i \in [1..t]$, T_i is a *timestamp* and $T_1 < T_2 < \ldots < T_t$. The *spatial dimension* is associated with a domain of values denoted by $dom(D_{\mathrm{S}}) = \{Z_1, Z_2, \ldots Z_l\}$ where $\forall i \in [1..l]$, Z_i is a *zone*. We define a neighborhood relationship on $dom(D_{\mathrm{S}})$, which is denoted by *Neighbor*, as:

$$\begin{cases} \textit{Neighbor}(Z_i, Z_j) = \textit{true}, & \text{if zones } Z_i \text{ and } Z_j \text{ are nearby} \\ \textit{Neighbor}(Z_i, Z_j) = \textit{false}, & \text{otherwise} \end{cases}$$

Each dimension D_{A_i} ($\forall i \in [1..p]$) in the set of *analysis dimensions* D_{A} is associated with a domain of values denoted by $dom(A_i)$. In these domains, the values can be ordered or not.

To illustrate these definitions, we use a weather database as an example. Table 2, represents weather in three zones on three consecutive days, in which, temperature (Temp), precipitation (Prec), wind speed (Wind) and gusts in Km/h are listed. The three zones are associated by a neighborhood relationship described in Fig. 1.

In Table 2, $D_{\mathrm{T}} = \{\textit{Date}\}$, $D_{\mathrm{S}} = \{\textit{Zone}\}$ and $D_{\mathrm{A}} = \{\textit{Temp, Prec, Wind, Gusts}\}$. The domain of the temporal dimension is $dom(D_{\mathrm{T}}) = \{$12/22/13, 12/23/13, 12/24/13$\}$ with 12/22/13 $<$ 12/23/13 $<$ 12/24/13. The domain of spatial dimension is $dom(D_{\mathrm{S}}) = \{Z_1, Z_2, Z_3\}$ with $\textit{Neighbor}(Z_1, Z_2) = \textit{true}$, $\textit{Neighbor}(Z_1, Z_3) = \textit{true}$ and $\textit{Neighbor}(Z_2, Z_3) = \textit{false}$. Finally, for the analysis dimensions *Temp* and *Gusts*, the domains are $dom(\textit{Temp}) = \{T_m, T_l, T_s\}^2$ and $dom(\textit{Gusts}) = \{55, 75\}$, respectively.

3.2. Spatio-sequential patterns

Definition 1 (Item and Itemset). Let I be an *item*, a literal value for the dimension D_{A_i}, $I \in dom(D_{\mathrm{A}_i})$. An *itemset*, $IS = (I_1 I_2 \ldots I_n)$ with $n \leqslant p$, is a non empty set of *items* such that $\forall i, j \in [1..n], \exists k, k' \in [1..p]$, $I_i \in dom(D_{A_k})$, $I_j \in dom(A_{D_{A_{k'}}})$ and $k \neq k'$.

All items in an itemset are associated with different dimensions. An itemset with k items is called k-itemset.

We define the In relationship between *zones* and *itemsets* which describes the occurrence of itemset IS in zone Z at time t in the database DB: $In(IS, Z, t)$ is true if IS is present in DB for zone Z at

[2] s = strong, m = medium, l = low.

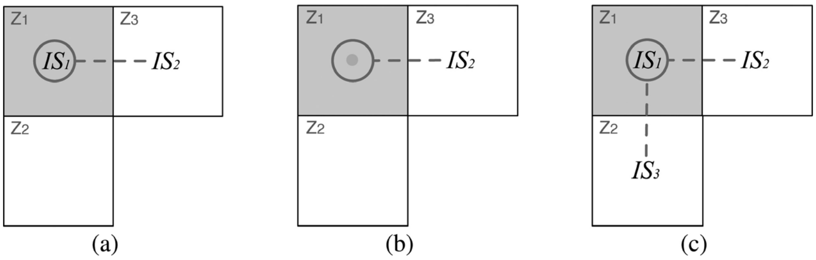


Fig. 2. Graphical representation of spatial itemsets (a) $IS_1 \cdot IS_2$ (b) $\theta \cdot IS_2$ (c) $IS_1 \cdot [IS_2; IS_3]$.

time t. In our example, consider the itemset $IS = (T_m P_m W_l)$ then $In(IS, Z_1, 12/23/13)$ is *true* as the itemset $(T_m P_m W_l)$ occurs for zone Z_1 on 12/23/13 (see Table 2).

Now, we define the notion of *interaction* between two nearby zones.

Definition 2 (Spatial itemset). Let IS_i and IS_j be two itemsets, we say that IS_i and IS_j are spatially close iff $\exists Z_i, Z_j \in dom(D_S)$, $\exists t \in dom(D_T)$ such that $In(IS_i, Z_i, t) \wedge In(IS_j, Z_j, t) \wedge$ *Neighbor*(Z_i, Z_j) is true. A pair of itemsets IS_i and IS_j that are spatially close, is called a *spatial itemset* and denoted by $I_{ST} = IS_i \cdot IS_j$.

To facilitate notations, we introduce a n-ary group operator for itemsets to be assigned by the operator $\cdot$ (*near*), denoted by *[]*. The θ symbol represents the *absence* of itemsets in a zone. Figure 2 shows the three types of spatial itemsets that we can build with the proposed notations. In Fig. 2, the shaded square represents the studied area, each IS represents an itemset and the dotted lines represent spatial neighborhood.

For instance, the spatial itemset $I_{ST} = (T_m \cdot P_m W_l)$ describes that events T_m and $P_m W_l$ occur in neighboring zones at the same time. The spatial itemset $I_{ST} = (\theta \cdot [T_m; P_l])$ indicates that T_m and P_l occur in two different zones neighbor to a zone where no event appears.

Definition 3 (Association between zone, spatial itemset and time). Let $I_{ST} = IS_i \cdot IS_j$ be a spatial itemset, $Z \in dom(D_S)$ be a zone and $t \in dom(D_T)$ be a timestamp, we define the relation *Verify* that represents the occurrence of the spatial itemset I_{ST} in Z at time t as follows:

$$\begin{cases} \text{Verify}(I_{ST}, Z, t) = true & \text{if } In(IS_i, Z, t) = true, \text{ and } \exists Z' \in dom(D_S) \text{ such} \\ & \text{that } \textit{Neighbor}(Z, Z') = true \text{ and } In(IS_j, Z', t) = true \\ \text{Verify}(I_{ST}, Z, t) = false, & \text{otherwise} \end{cases}$$

Consider the spatial itemset $I_{ST} = (P_m W_m \cdot 75)$ then *Verify*$(I_{ST}, Z_1, 22/12/2013) = true$ indicates that itemset $(P_m W_m)$ occurs in zone Z_1 and (75) in a neighbor of Z_1 at time 12/22/2013 (see Table 2).

Definition 4 (Inclusion of spatial itemset). A spatial itemset $I_{ST} = IS_i \cdot IS_j$ is *included* in another spatial itemset I'_{ST} = $IS'_k \cdot IS'_l$, denoted by $I_{ST} \subseteq I'_{ST}$, iff $IS_i \subseteq IS'_k$ and $IS_j \subseteq IS'_l$.

The spatial itemset $I_{ST} = (T_l P_m \cdot W_s)$ is *included* in the spatial itemset $I'_{ST} = (T_l P_m \cdot W_s 55)$ because $(T_l P_m) \subseteq (T_l P_m)$ and $(W_s) \subseteq (W_s 55)$.

We now define the notion of zones *evolution* according to their spatial neighborhood relationship.

Definition 5 (Spatial Sequence). A spatial sequence or simply *2S* is an ordered list of spatial itemsets, denoted by $s = \langle I_{ST_1} I_{ST_2} \ldots I_{ST_m} \rangle$ where I_{ST_i}, $I_{ST_{i+1}}$ satisfy the constraint of temporal sequentiality for all $i \in [1..m-1]$.

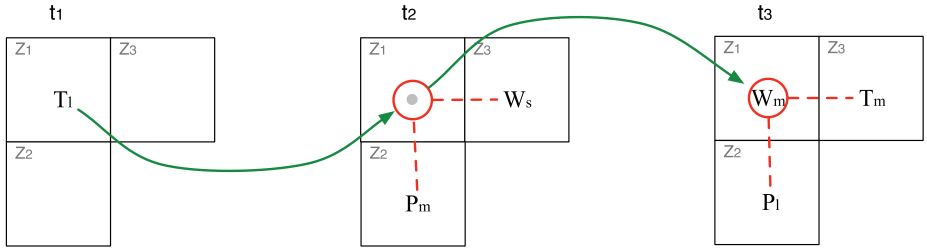


Fig. 3. Spatiotemporal dynamic of sequence $\langle (T_l)(\theta \cdot [P_m; W_s])(W_m \cdot [P_l; T_m])\rangle$ that describes the evolution of the zone Z_1. (Colours are visible in the online version of the article; http://dx.doi.org/10.3233/IDA-160806)

A 2S $s = \langle (T_l)(\theta \cdot [P_m; W_s])(W_m \cdot [P_l; T_m])\rangle$ is illustrated in Fig. 3 for the zone $Z1$, where the arrows represent the temporal dynamics and the dotted lines represent the proximity relationship.

A generalization relationship (or specialization) between 2S's is defined as follows:

Definition 6 (Inclusion of 2S). A 2S represented by $s = \langle I_{ST_1} I_{ST_2} \ldots I_{ST_m}\rangle$ is more specific than a 2S $s' = \langle I'_{ST_1} I'_{ST_2} \ldots I'_{ST_n}\rangle$, denoted by $s \preceq s'$, if there exists $j_1 \leqslant \ldots \leqslant j_m$ such that $I_{ST_1} \subseteq I'_{ST_{j_1}}, I_{ST_2} \subseteq I'_{ST_{j_2}}, \ldots, I_{ST_m} \subseteq I'_{ST_{j_m}}$.

A 2S $s = \langle (T_l P_m \cdot P_l W_s)(55)\rangle$ is included in the 2S $s' = \langle (T_l P_m \cdot P_l W_s)(55 \cdot W_s)\rangle$ because $(T_l P_m \cdot P_l W_s) \subseteq (T_l P_m \cdot P_l W_s)$ and $(55) \subseteq (55 \cdot W_s)$.

Definition 7 (Prefix and suffix of 2S). We define the function *prefix*$(S, N) \to S$ where S is a set of 2S's, N is a set of positive integers, and *prefix*$(s, k) = s[1 : k]$. In other words, *prefix*(s, k) returns the first k *items*, i.e., the *prefix* of s. In the same way, we define the function *suffix*$(S, S) \to S$ where S is a set of 2S's. Let $s \in S$ with n items and $s' \in S$ with m items, if s' is a prefix of s then the *suffix*$(s, s') = s[m + 1 : n]$ returns the 2S of $n - m$ last items of the sequence s prefixed by s'.

Let the spatial sequence $s_1 = \langle (T_l P_m \cdot P_l W_s)(55)\rangle$ and $s_2 = \langle (T_l P_m\rangle$, then, the suffix s_1 compared to prefix s_2 is $\langle (\_ \cdot P_l W_s)(55)\rangle$.

The spatial sequences 2S can be stored in a sequence database seqDB.

Definition 8 (Projection of a sequence database seqDB). Let s be a 2S present in the sequence database seqDB. The s-projected database, denoted by $seqDB|_s$ is a set of suffixes of spatial sequences in seqDB prefixed by s.

4. Pruning measures

In this section, we introduce two pruning measures to filter interesting spatial sequences in the mining process. The first one is an adaptation of the support used in sequential pattern mining [2], while the second one is an adaptation of the cascading participation index used [17]. To illustrate definitions proposed in this section, we use a database of sequences. The spatiotemporal database presented in Table 2 can be transformed in a database of sequences by organizing the events of the analysis dimension (D_A) by date and by zone. Hence, sequences shown in Table 3 were built from our spatiotemporal database.

Hereafter, each zone $Z_i \in dom(D_S)$ will be represented by their sequence S_i, for example, the sequence $S_1 = \langle (T_l P_m W_m)(T_m P_m W_l)(T_l P_m W_m 55)\rangle$ represents the events occurred in Z_1.

Table 3
Sequences per zone

Zone	Sequences
Z_1	$S_1 = \langle (T_l P_m W_m)(T_m P_m W_l)(T_l P_m W_m 55) \rangle$
Z_2	$S_2 = \langle (T_m P_m W_m)(T_l P_m W_l)(T_l P_l W_m) \rangle$
Z_3	$S_3 = \langle (T_l P_m W_s 75)(T_m P_s W_l)(T_m P_s W_s 55) \rangle$

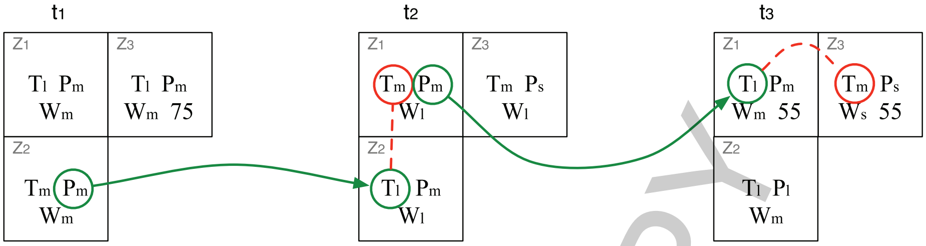


Fig. 4. Spatiotemporal dynamic of S2P $\langle (P_m)(T_l \cdot T_m) \rangle$, that appears twice: in Z_2 and in Z_1. (Colours are visible in the online version of the article; http://dx.doi.org/10.3233/IDA-160806)

4.1. *Support of 2S*

In sequence mining literature, a common example is the basket data where the database is composed of a set of sequences corresponding to customers transactions. Each transaction consists of customer-id, transaction time and the items bought in the transactions. The absolute support for a sequence is defined as the number of transactions in database which contains the sequence [2]. Our spatiotemporal database is similar to a sequence database, since information of a zone at different times can be viewed as a sequence. The main difference is the neighborhood relationship associated to the spatial dimension. As a consequence, we define a new *absolute support* for spatial sequences as the number of zones containing the studied sequence that satisfies the proximity constraints of the spatial itemsets. More formally, this measure can be defined as follow:

Definition 9 (Absolute support). Let $s = \langle I_{ST_1} I_{ST_2} \dots I_{ST_n} \rangle$ be a 2S, the absolute support of s is defined as:

$$supp_{abs}(s) = |\{Z \in dom(D_S) | \forall i \in [1..n], \exists T_i \in dom(D_T), \\ \text{and } Verify(I_{ST_i}, Z, T_i) = true\}| \tag{1}$$

In the same way, we define the *relative support* for a 2S as the fraction of total zones which supports a 2S. More formally:

Definition 10 (Relative support). Let $s = \langle I_{ST_1} I_{ST_2} \dots I_{ST_n} \rangle$ be a 2S, the relative sequential support of s represented by $supp_{rel}(s)$ or simply $supp(s)$, is defined as:

$$supp_{rel}(s) = supp(s) = \frac{supp_{abs}(s)}{|dom(D_S)|} \tag{2}$$

For instance, consider the neighborhood relationship shown in Fig. 1 and sequences presented in Table 3. Each sequence represents a change in events per zone (see Table 2). Let $s = \langle (P_m)(T_l \cdot T_m) \rangle$ be a 2S, the relative support of s is $2/3$ (i.e. s appears in two of three zones). The dynamic and support of s are illustrated in Fig. 4. The arrows represents the temporal changing and dotted lines represents the spatial neighborhood relationship.

4.2. Spatiotemporal participation index

The previous support does not take into account the participation of an item in a spatio-sequential pattern. It does not evaluate occurrences of a pattern appearing in the same zone at different times. To highlight these aspects, we propose a new anti-monotonic prune measure called *Spatiotemporal Participation Index (STPi)*. This measure is an adaptation of the cascading participation index [17] that is a combination of two measures: (1) *the spatial participation index*, which takes into account the number of zones supporting the pattern, and; (2) *the temporal participation index*, which takes into account the number of occurrences of a pattern in a zone at different times.

First, we define the spatial participation ratio and the spatial participation index.

Let s be a 2S and I be an item of s. The *spatial participation ratio* of I in s, denoted by $SPr(s, I)$ is the number of zones which contain s divided by the number of zones where the item I appears in the whole database:

$$SPr(s, I) = \frac{supp_{abs}(s)}{supp_{abs}(I)} \tag{3}$$

Let s be a 2S, the *spatial participation index* of s, denoted by $SPi(s)$ is the minimum of *spatial participation ratio*:

$$SPi(s) = MIN_{\forall I \in dom(A_i), I \in s}\{SPr(s, I)\} \tag{4}$$

Then, we define the temporal participation ratio and the temporal participation index.

Let s be a 2S, let I be an item of s and let S_i be a sequence that represents events occurred in Z_i which supports s. The *temporal participation ratio* of I in s denoted by $TPr(s, I, S_i)$ is the number of instances of I participating in an occurrence of s divided by the total number of instances of I in sequence S_i supporting s:

$$TPr(s, I, S_i) = \frac{\textit{nbInstances}(I) \textit{ participating in an occurrence of } s}{\textit{nbInstances}(I) \textit{ in } S_i \textit{ supporting } s} \tag{5}$$

Let s be a 2S, the *temporal participation index* of s, denoted by $TPi(s)$ is the minimum of *temporal participation ratio* calculated for each sequence in database:

$$TPi(s) = MIN_{S_i, i \in dom(D_S)}\{MIN_{I \in dom(A_i), I \in s}\ \{TPr(s, I, S_i)\}\} \tag{6}$$

Now, using previous definitions, we define the Spatiotemporal Participation Index as follow:

Definition 11 (Spatiotemporal Participation Index). We define the *spatiotemporal participation index* of a spatial sequence s, $STPi(s)$, as the product of two measures presented above:

$$\textit{STPi}(s) = SPi(s) \times TPi(s) \tag{7}$$

For example, consider the same 2S presented in Section 4.1, i.e., $s = \langle(P_m)(T_l \cdot T_m)\rangle$. Also, consider the neighborhood relationship shown in Fig. 1 and sequences presented in Table 3, the *spatiotemporal participation index* for s is $1/3$, as shown below:
Initially, we compute the *spatial participation index*:

$$SPi = \textit{MIN}\{SPr(s, P_m), SPr(s, T_l), SPr(s, \cdot T_m)\} = \textit{MIN}\left\{\frac{2}{3}, \frac{2}{3}, \frac{2}{3}\right\} = \frac{2}{3}$$

Next, the *temporal participation index* is computed:

$$\begin{aligned} TPi &= \mathit{MIN}_{\forall S_i, i \in dom(D_S)} \left\{ \mathit{MIN} \left\{ TPr(s, P_m), TPr(s, T_l), TPr(s, \cdot T_m) \right\} \right\} \\ &= \mathit{MIN} \left\{ \left\{ \frac{2}{3}, \frac{1}{1}, \frac{2}{4} \right\}, \left\{ \frac{1}{2}, \frac{1}{1}, \frac{1}{1} \right\}, \left\{ \right\} \right\} \\ &= \mathit{MIN} \left\{ \left\{ \frac{1}{2} \right\}, \left\{ \frac{1}{2} \right\} \right\} = \frac{1}{2} \end{aligned}$$

Finally, two results obtained above are replaced in the equation of *spatiotemporal participation index*:

$$\mathit{STPi}(s) = SPi(s) \times TPi(s) = \frac{2}{3} \times \frac{1}{2} = \frac{1}{3}$$

The main challenge involved in the spatio-sequential patterns mining problem is: let σ be a minimum threshold set by the user, a spatial sequence 2S satisfying $\mathit{STPi}(2S) \geqslant \sigma$ is a frequent sequence called *Spatio-Sequential Patterns* or simply S2P. It is important to notice that we can use the support *supp* instead of the Spatiotemporal Participation Index *STPi* to validate a 2S as a frequent sequence.

4.3. Discusion

As has been discussed [12], one of the most important task in data mining process is how the frequency of a sequence is estimated. This issue will be addressed in this discussion.

In the literature, there are several measures for evaluating patterns discovered by the data mining process (for an overview, see, e.g., [7,16]). For instance, [17], the authors propose *the cascade participation index (CPI)*, which is defined as the minimum number of instances of an event-type participating in a pattern divided by the number of instances of event-type in all the database. To estimate the CPI, the authors count "globally" the occurrences of a pattern in the whole database.

In this paper, we propose an adaptation of the aforementioned measure taking into account two aspects. First, in TPi, we propose to evaluate the occurrences of a spatial sequence 2S "locally" (as opposed to globally), counting the number of event occurrences in a sequence where the pattern appears (as opposed to whole database). Second, as detailed in [25], the presence of *null-sequences*[3] is critically important in the process of computing an interesting measure. Our proposition takes into account this constraint evaluating only the sequences containing the studied spatial sequence 2S.

5. Extraction of spatio-sequential patterns

In this section, we propose two algorithms associated with the two most common strategies in data mining to find spatio-sequential patterns in a spatiotemporal database. In a first step, we use the breadth-first search strategy (BFS) and we propose an algorithm BFS-S2PMiner. In a second step, we propose an algorithm DFS-S2PMiner based on a depth-first search strategy (DFS).

[3] A null-sequence w.r.t. an item I, where $I \in dom(A_i)$, is a sequence in a sequence database that does not contain the item I.

Table 4
Generation of candidate patterns

Sequence α	Sequence β	If	Candidate pattern γ
$\langle(\rho X)\rangle$	$\langle(\rho Y)\rangle$	$X < Y$	$\langle(\rho XY)\rangle$
$\langle(\rho X)\rangle$	$\langle(\rho)(Y)\rangle$		$\langle(\rho X)(Y)\rangle$
$\langle(\rho)(X)\rangle$	$\langle(\rho)(Y)\rangle$	$X < Y$	$\langle(\rho)(XY)\rangle$
$\langle(\rho X)\rangle$	$\langle(\rho \cdot Y)\rangle$		$\langle(\rho X \cdot Y)\rangle$
$\langle(\rho X)\rangle$	$\langle(\rho)(\theta \cdot Y)\rangle$		$\langle(\rho X)(\theta \cdot Y)\rangle$
$\langle(\rho \cdot X)\rangle$	$\langle(\rho)(Y)\rangle$		$\langle(\rho \cdot X)(Y)\rangle$
$\langle(\rho \cdot X)\rangle$	$\langle(\rho \cdot Y)\rangle$	$X < Y$	$\langle(\rho \cdot [X;Y])\rangle$
$\langle(\rho)(\theta \cdot X)\rangle$	$\langle(\rho)(\theta \cdot Y)\rangle$	$X < Y$	$\langle(\rho)(\theta \cdot [X;Y])\rangle$

Table 5
Multi-sets of sequences and neighboring sequences

Sequences	Neighboring sequences
$S_1 = \langle(T_l P_m \boldsymbol{W_m})(T_m P_m W_l)(T_l P_m W_m 55)\rangle$	$S_2 = \langle(T_m P_m \boldsymbol{W_m})(T_l P_m W_l)(T_l P_l W_m)\rangle$
	$S_3 = \langle(T_l P_m W_s 75)(T_m P_s W_l)(T_l P_s W_s 55)\rangle$
$S_2 = \langle(T_m P_m \boldsymbol{W_m})(T_l P_m W_l)(T_l P_l W_m)\rangle$	$S_1 = \langle(T_l P_m \boldsymbol{W_m})(T_m P_m W_l)(T_l P_m W_m 55)\rangle$
$S_3 = \langle(T_l P_m W_s 75)(T_m P_s W_l)(T_l P_s W_s 55)\rangle$	$S_1 = \langle(T_l P_m \boldsymbol{W_m})(T_m P_m W_l)(T_l P_m W_m 55)\rangle$

5.1. Level-wise approach

In level-wise algorithms (generate-test-prune), the approach first extracts the frequent items, then, for each iteration k, the algorithm generates a set of candidates from frequent patterns generated at iteration $k - 1$. At the end of each iteration, a pruning phase based on an anti-monotone property, is done to limit the number of candidate patterns generated. All candidates are tested and only those which are frequent, are used to generate candidate patterns with k items during the next iteration. The algorithm stops when the set of candidates is empty. There are many algorithms based on this approach, among which we find *Apriori* [1] *GSP* [22] or *SPADE* [27]. We propose in this section, a new algorithm derived from the algorithm *SPADE* introduced by Zaki [27].

Initially, our algorithm extracts frequent items (lines 1 to 8 in Algorithm 1) to construct the set F_1. From this set F_1, we construct the candidates of iteration 2. In general, at each level, we construct set of patterns with $k + 1$ items from those with k items. Note that in each iteration k, the patterns generated have k items. For example, consider two spatial sequences $\langle(XY \cdot Z)\rangle$ and $\langle(XY)(Y)\rangle$, both composed of three items. The prefix of these two patterns is $\langle XY\rangle$ then we can generate the pattern $\langle(XY \cdot Z)(Y)\rangle$ with four items.

More formally, let the sequences α and β having the same prefix ρ. In the candidate generation step, we generate candidate patterns γ having the same prefix following the rules described in Table 4.

In the pruning step, the algorithm prunes candidate patterns which have an unfrequent $k-1$ sub-pattern (line 15). For example, the pattern $\langle(XYZ)\rangle$ is not generated if one of its sub-patterns (e.g., $\langle(XY)\rangle$) is not frequent.

Finally, once the pruning step is complete, the algorithm tests each candidate patterns (line 16) and uses the k-frequent patterns discovered to begin the next iteration. Algorithm 1 is the pseudo-code for this proposal.

We illustrate our proposal using the neighborhood relationship described in Fig. 1, a minimal threshold $\sigma = 2/3$ and a sequence database described in Table 5. This table is divided in two parts: the first stores the sequences that represent the evolution over time in each zone of the spatiotemporal database, and, the second one, stores sequences associated to their neighboring zones.

Algorithm 1 BFS-S2PMiner

```
Require: A sequence database $seqDB$, a neighborhood relationship $L$ and a minimal threshold $\sigma$
Ensure: A set of frequent spatio-sequential patterns
 1: for all $i \subseteq dom(A_i)$ where $i \in [1..p]$ do
 2:   if $STPi(i) \geqslant \sigma$ then
 3:     $F_1 \leftarrow F_1 \cup \{i\}$
 4:   end if
 5:   if $STPi(\cdot i) \geqslant \sigma$ then
 6:     $F_1 \leftarrow F_1 \cup \{\cdot i\}$
 7:   end if
 8: end for
 9: $k \leftarrow 1$
10: while $F_k \neq \emptyset$ do
11:   $F_{k+1} \leftarrow \emptyset$
12:   for all $\alpha \in F_k, \beta \in F_k$ do
13:     if $prefix(\alpha) = prefix(\beta)$ then
14:       $\gamma \leftarrow union(\alpha, \beta)$
15:       if $AllSubSeqFreq(F_k, \gamma)$ then
16:         if $STPi(\gamma) > \sigma$ then
17:           $F_{k+1} \leftarrow F_{k+1} \cup \{\gamma\}$
18:         end if
19:       end if
20:     end if
21:   end for
22:   $k \leftarrow k + 1$
23: end while
24: return $F_1 \cup F_2 \cup \ldots \cup F_k$
```

Lines 1 to 8 of our algorithm calculates the set F_1 of frequent items, by checking that the minimal threshold σ is achieved. F_1 is composed of all items that appear at least twice in sequences associated with zones and those that appear at least twice in their neighboring sequences.

In our example (see Table 5), the item W_m appears twice in sequences S_1 and S_2 and item $\cdot W_m$ appears three times: in the neighborhood of S_1 (i.e., in S_2) in the neighborhood of S_2 (i.e., in S_1) and in the neighborhood of S_3 (i.e., in S_1). Finally, F_1 is composed of:

$$F_1 = \{T_l : 3, W_m : 2, W_l : 3, P_m : 3, T_m : 3, 55 : 2, \cdot T_l : 3, \cdot W_m : 3, \cdot W_l : 3, \cdot P_m : 3,$$
$$\cdot\, T_m : 3, \cdot 55 : 3\}$$

The number associated with these items is the frequency of occurrence of these items in Table 5.

We can notice that item $\cdot T_l$ appears four times in neighboring sequences, but its support is three. In fact, item $\cdot T_l$ appears once in the neighborhood of S_2, once in the neighborhood of S_3 and twice in the neighborhood of S_1. However, in this latter case, the support of $\cdot T_l$ is counted only one time because it appears in the same neighboring areas of S_1.

From the set F_1, we construct the set F_2 of spatio-sequential patterns prefixed by frequent patterns found in the previous step. In general, we proceed as follows: if two patterns have the same prefix (line

13), we generate a candidate pattern γ with $k+1$ items which is obtained by the *union* of two frequent patterns containing k items according to the cases presented in Table 4 (line 14).

Examples of candidates with 2 items generated from the set F_1 are $(T_l)(T_l)$, (T_lW_m), (T_lW_m), ..., $(T_l \cdot T_l)$, $(T_l)(\theta \cdot T_l)$, ... In this set of two candidates, we only keep those whose subsets of size 1 are frequent (line 15).

Then, at line 16, the algorithm checks whether the candidate patterns of size 2 are frequent. For example, the pattern $(T_m)(W_f)$ is frequent because it appears twice in sequences S_1 and S_2 (see Table 5).

Finally, we obtain the set of frequents spatio-sequential patterns composed of 2 items:

$$F_2 = \{(T_l)(T_l) : 3, (T_l \cdot T_l) : 3, (T_l)(\theta \cdot T_l) : 2, (T_lW_m) : 2, (W_m \cdot W_m) : 3, (P_m)(P_m) : 2,$$
$$(P_m \cdot P_m) : 3, (P_m)(\theta \cdot P_m) : 3, \ldots\}$$

By performing previous steps *iteratively*, the algorithm continues until there is no candidate that can be generated. Then, it returns the set of frequent spatio-sequential patterns.

SPADE is considered as an *Apriori-like* algorithm. The computation complexity of BFS-S2PMiner is, in the worst-case, $\mathcal{O}(TN(2M)^N)$ where T is the number of transactions on the database ($|dom(D_S)|$), N is the number of different items in the database and M is the maximum number of neighboring zones.

5.2. *Depth-first approach*

In this section, we propose an algorithm called DFS-S2PMiner to extract spatio-sequential patterns considering both spatial and temporal aspects. DFS-S2PMiner adopts a depth-first-search strategy based on successive projections of the database such as FP-Growth [10] and Prefixspan [18]. Specifically, this algorithm is based on the *pattern-growth* strategy used [10]. The principle of this approach is to extract frequent patterns without a candidate generation step. Indeed, the level-wise algorithm stage can be very expensive due to the large number of candidate patterns that can be generated. Moreover, it requires repeated scanning of the database and checking of the support of a large number of patterns. Strategies such as *pattern-growth* avoid this by using a *divide and conquer* approach. This approach recursively creates a projected database (cf. Definition 8), associates it with a fragment of frequent pattern, and "mines" each projected database separately. The frequent patterns are extended progressively along a depth-first exploration of the search space.

The Algorithm 2 describes our recursive algorithm DFS-S2PMiner. First, the set of frequent items I and $\theta \cdot I$, denoted by F_1, is extracted from the projected database $seqDB|_\alpha$ (line 1 of Algorithm 2). These items constitute extensions of the sequence α. Note that in the first recursive call, $seqDB|_\alpha$ corresponds to the initial database $seqDB$ (since $\alpha = \{\}$). Then, for each of these items $X \in F_1$, we extend the spatio-sequential pattern α with X (lines 3 and 4). Two types of extension are possible: 1) adding X to the last spatial itemset of the sequence α (line 3) or 2) inserting X after (i.e., the next time) the last spatial itemset of α (line 4). We check the interestingness measure for these two spatio-sequential patterns (lines 5 and 9) and record frequent ones in the set of solutions F (lines 6 and 10). For each frequent pattern, the algorithm then performs another projection of the database using $seqDB|_\alpha$ and recursively extends the pattern by invoking again the algorithm (lines 7 and 11). The algorithm stops when no more projections can be generated.

We use our running example (Table 5 and Fig. 1) with $\sigma = 2/3$ to illustrate this algorithm.

Table 6
Projected database of $\langle(P_m)\rangle$

Sequences	Neighboring sequences
$S_1 = \langle(\_\boldsymbol{W_m})(T_m\boldsymbol{P_m}W_l)(T_l\boldsymbol{P_m}\boldsymbol{W_m}55)\rangle$	$S_2 = \langle(\_W_m)(T_lP_mW_l)(T_lP_lW_m)\rangle$ $S_3 = \langle(\_W_s75)(T_mP_sW_l)(T_lP_sW_s55)\rangle$
$S_2 = \langle(\_\boldsymbol{W_m})(T_l\boldsymbol{P_m}W_l)(T_lP_l\boldsymbol{W_m})\rangle$	$S_1 = \langle(\_W_m)(T_mP_mW_l)(T_lP_mW_m55)\rangle$
$S_3 = \langle(\_W_s75)(T_mP_sW_l)(T_lP_sW_s55)\rangle$	$S_1 = \langle(\_W_m)(T_mP_mW_l)(T_lP_mW_m55)\rangle$

Algorithm 2 DFS-S2PMiner

```
– Main routine
Require: A sequence database $seqDB$ and a user-defined threshold $\sigma$
Ensure: A set of frequent spatio-sequential patterns $F$
  $\alpha \leftarrow \{\}$
  $Prefix\text{-}growthST(\alpha, \sigma, seqDB|_\alpha, F)$

– Prefix-growthST$(\alpha, \sigma, seqDB|_\alpha, F)$

Require: a spatio-sequential pattern $\alpha$, the user-defined threshold $\sigma$, the projection $seqDB|_\alpha$ of the
    sequence database on $\alpha$, and $F$ a set of frequent spatiotemporal patterns;
 1. $F_1 \leftarrow$ {a set of frequents items $I$ and $\theta \cdot I$ on $seqDB|_\alpha$, with $I \in \bigcup_{i\in[1..p]} dom(D_{A_i})$}
 2. for all $X \in F_1$ do
 3.   $\beta \leftarrow \alpha X$
 4.   $\delta \leftarrow \alpha(X)$
 5.   if $STPi(\delta) \geqslant \sigma$ then
 6.     $F \leftarrow F \cup \delta$
 7.     Prefix-growthST$(\delta, \sigma, seqDB|_\delta, F)$
 8.   end if
 9.   if $STPi(\beta) \geqslant \sigma$ then
10.     $F \leftarrow F \cup \beta$
11.     Prefix-growthST$(\beta, \sigma, seqDB|_\beta, F)$
12.   end if
13. end for
```

Iteration 1 ($\alpha = \{\}$)

The first step concerns the extraction of frequent items and frequent spatial items from $seqDB$ (line 1 of Algorithm 2), let:

$$F_1 = \{P_m : 3, T_m : 3, W_m : 2, W_l : 3, T_l : 3, 55 : 2, \theta \cdot T_m : 3, \theta \cdot P_m : 3, \theta \cdot W_m : 3,$$
$$\theta \cdot W_l : 3, \theta \cdot T_l : 3, \theta \cdot 55 : 3\}$$

Next, the current sequence α is extended (lines 3 and 4). Thereafter, in lines 5, 6 and 9, 10, we process the STPi and record the solutions respectively.

For each frequent item I and $\theta \cdot I$, the algorithm calculates the corresponding projection of the database (lines 7 and 11). For example, for the frequent item P_m, we obtained the following projection (see Table 6). Each of these projected databases are used in a recursive call to find its frequent super-sequences.

Iteration 2 ($\alpha = \langle(P_m)\rangle$)

Table 7
Projected database of $\langle(P_m)(W_m)\rangle$

Zones	Sequences	Neighbors	Neighboring sequences
Z_1	$S_1 = \langle(T_mP_mW_l)(T_lP_mW_m55)\rangle$	Z_2	$S_2 = \langle(T_l\boldsymbol{P_m}W_l)(T_l\boldsymbol{P_m}W_m)\rangle$
		Z_3	$S_3 = -$
Z_2	$S_2 = \langle(T_lP_mW_l)(T_lP_lW_m)\rangle$	Z_1	$S_1 = \langle(T_m\boldsymbol{P_m}W_l)(T_l\boldsymbol{P_m}W_m55)\rangle$
Z_3	$S_3 = \emptyset$	Z_1	$S_1 = \langle(T_m\boldsymbol{P_m}W_l)(T_l\boldsymbol{P_m}W_m55)\rangle$

The first recursive call will build the super-sequences with the prefix $\langle(P_m)\rangle$ from the projected database of Table 6. Specifically, the algorithm will find frequent items in the projected database (line 1) and extend $\langle(P_m)\rangle$ (lines 3, 4). The frequent items obtained from $seqDB|_{\langle(P_m)\rangle}$ are:

$$\{W_m : 2, T_m : 2, P_m : 2, W_l : 3, T_l : 3, 55 : 2, \theta \cdot W_m : 3, \theta \cdot T_l : 3, \theta \cdot P_m : 3, \theta \cdot W_l : 3,$$
$$\theta \cdot T_m : 3, \theta \cdot 55 : 3\}$$

The first frequent item found is $\langle W_m\rangle : 2$. Therefore, we can build two spatial sequences: $\langle(P_mW_m)\rangle$ (line 3) and $\langle(P_m)(W_m)\rangle$ (line 4).

In lines 5, 6 and 9, 10, we process the STPi and record the solutions. For instance, the spatio-sequential pattern $\langle(P_m)(W_m)\rangle$ with $STPi = 2/3$ is frequent (line 9).

Later, a projection and a recursive call are performed (lines 7 and 11). Thus, the algorithm uses this pattern to make a new projection (see Table 7) and to recursively search all frequent super-sequences with the prefix $\langle(P_m)(W_m)\rangle$.

Iteration 3 $(\alpha = \langle(P_m)(W_m)\rangle)$

The frequent items obtained for $seqDB|_{\langle(P_m)(W_m)\rangle}$ are:

$$\{W_m : 2, P_m : 2, W_l : 2, T_l : 2, \theta \cdot W_m : 3, \theta \cdot T_l : 3, \theta \cdot P_m : 3, \theta \cdot W_l : 3, \theta \cdot 55 : 3\}.$$

Next, we extend the current sequence α (lines 3, 4). For example, the spatial item $\theta \cdot P_m : 3$ is one of the frequent items. In this case, the algorithm builds the spatio-sequential pattern $\langle(P_m)(W_m)(\theta \cdot P_m)\rangle$. This pattern is frequent with a $STPi = 1$ because $\langle\theta \cdot P_m\rangle$ appears in all times and zones (see Table 7).

Finally, a projection and a recursive call (lines 7 and 11) are performed. When all frequent items are projected, the algorithm goes through another *branch* of the search space, i.e., patterns beginning with $\langle(T_m)\rangle$ (see set F_1).

The algorithm thus proceeds generally in the same way whether items are spatial or not. The main difference is how to identify a frequent sequence. The support of a spatial item is the number of zones where the item occurs at least once in their neighborhood (so we have $\theta \cdot W_l : 3$ in Table 6). Notice that when the algorithm extends a pattern of type $\langle(I_{ST_1})(I_{ST_2})\dots(I_{ST_k} \cdot X)\rangle$ with a common item $\theta \cdot Y$, the operator of $n\text{-}ary$ group is used to represent the sequence as $\langle(I_{ST_1})(I_{ST_2})\dots(I_{ST_k} \cdot [X;Y])\rangle$.

The worst-case computation complexity of DFS-S2PMiner is $\mathcal{O}((2NM)^L)$ where N is the number of different items in the database, M is the maximum number of neighboring zones, and L is the maximum length of all transactions. The constant 2 is introduced since each item can be added into transaction through either itemset or sequence extension.

6. Experiments and results

In order to evaluate the performance of our propositions, experiments were performed on two real spatiotemporal datasets with different relevant feature types (e.g., number of dengue cases, river pollution

Table 8
Characteristics of synthetic datasets

Dataset	Number of zones	Number of dates	Number of items
Graph10x50 (graph)	10	50	5
Graph10x70 (graph)	10	70	5
Graph20x70 (graph)	20	70	5
Graph20x100 (graph)	20	100	5
Grid20x100 (grid)	20	100	5

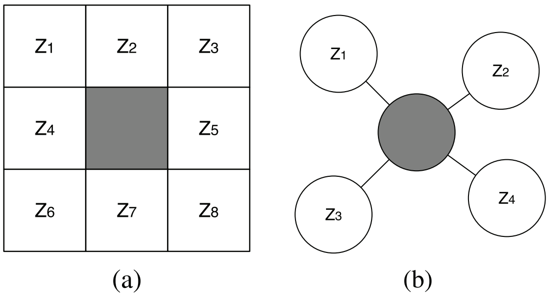


Fig. 5. Types of neighborhood relationships (a) grid type (b) graph type.

measure. . .) and also on synthetic datasets generated using empirical realistic values. In the early part of this section, we will describe the datasets, later, we analyze our results taking into account qualitative and quantitative aspects. This section ends with a performance and scalability study.

6.1. Dataset description

Proposed algorithms were tested using synthetic and real datasets described below.

6.1.1. Synthetic datasets

To obtain randomly generated data, we construct a configurable synthetic data generator with the following parameters: number of zones, type of neighborhood relationships, number of dates per zone and the average number of items per zone. This generator allows us to build two types of neighborhood relationships (see Fig. 5). The first one is used to represent the space as a grid where each square represents an area. An area will have eight neighbors. The second one is used to represent the space as a graph structure where vertices represent zones and edges represent neighborhood relationships. In the last case, the generator takes as input parameters the number of edges and the number of vertices. Large and different synthetic datasets were generated to evaluate our algorithms. Simulation parameters were chosen based on empirical realistic values.

As a summary, Table 8 shows the characteristics of the synthetic datasets that have been used for experimentations.

6.1.2. Dengue dataset

Dengue is a mosquito-borne infection (*Aedes aegypti*) that occurs in all tropical and subtropical regions of the planet. Currently, dengue viral infection has become an increasing global health concern with over two-fifths of the world's population at risk of infection. It is the most rapidly spreading vector borne disease, attributed to demographics, urbanization and environment changing. The severe dengue (formerly known as dengue hemorrhagic fever) was first identified in the fifties of last century during an outbreak in the Philippines and Thailand. Today, this disease affects most of the countries in Asia and Latin America (over 100 tropical and sub-tropical countries) and has become one of the leading causes of hospitalization and death among children in these regions. An estimated 500 000 people with severe dengue require hospitalization each year, a large proportion of whom are children. About 2.5% of those affected die.[4]

[4] Official website of World Health Organization.

Table 9
Attributes of dengue monitoring dataset

Attribute	Description
id_zone	Zone identification
Date	Record date
Precip	Precipitation in mm by comune
mean_wind	Mean wind strength in m/s by zone
mean_temper	Mean temperature in °C by zone
mean_humid	Mean humidity in % by zone
outdoor_deposit	Number of water deposits in outdoor communal areas (ponds, watersheds, fountains, etc.)
Graveyard	Number of cemeteries by zone
waste_container	Waste containers by zone
indoor_deposit	Number of water deposit in urban areas (drainage, ditches, outlets, standpipe inlets, etc.)
community_gather	Number of communal centers resembling people (schools, churches, universities, etc.)
ihre_index	Entomological high risk index
nb_cas_dengue	Number of dengue cases by zone

In the context of a collaboration between the University of New Caledonia, the Department of Health and Social Affairs of New Caledonia, the Pasteur Institute and the Institute of Research for Development, we have analyzed data associated with epidemiological monitoring of dengue. These data were collected in *Nouméa (New Caledonia)* in a territory divided into 81 zones covering 45.7 km^2. This spatial division was proposed by the Direction of Health and Social Affairs in New Caledonia. This dataset contains information associated to: population data, entomological data, meteorological data, urban planning data and medical data which are summarized in Table 9. Overall, the dengue dataset contains between 22 dates in average by zone and in average 22 items by date.

Population data: population data is related to population census and includes the code of zone, the number of inhabitant, the number of houses, the number of households, etc. This data was collected during two epidemic years (1996 and 2003) for each zone in Nouméa.

Entomological data: this data is associated to the characteristics of transmission vector (*Aedes aegypti*). This data includes *Breteau* Index (IB), Entomological High Risk Index (IHRE), data related to serotype, the positivity rate and other information used by the epidemiologists to analyze the disease. In this category, the data is retrieved by zone and the time granularity is monthly.

Meteorological data: this data comprises several weather signs such as precipitations (mm), wind strength (m/s), temperature (°C) and humidity (%). This data was recorded daily for each zone in Nouméa.

Urban planning data: Aedes aegypti mosquitoes live and breed in urban areas in close proximity to humans. The mosquito breeds in artificial containers (e.g., old tires, pot plant trays, urns or rainwater containers) that collect water and feeds almost exclusively on human blood. Considering these two aspects, we integrate: (1) in one hand, data associated to essential breeding ground for mosquitoes development as number of pools, number of greenhouses, number of basins or watersheds, fountains, and; (2) data related to places where people gather for social activities like, schools, churches, nurseries, etc.

Medical data: this data only comprises the number of dengue cases reported in Nouméa per day. Years 1996 and 2003 are the most important in terms of number of dengue cases. We will use only the data of 2003 which contains dengue cases.

In order to obtain categorical data (data separable into categories that are mutually exclusive), a discretization has been done to recode continuous data into categorical data by using the discretization method of *equi-width binning*. The data is stored according to three range of values within which they are classified: low, medium and high.

Table 10
Attributes of Saône river dataset

Attribute	Description
id_area	Zone identification
date	Date
ibgn	Standardized global biological index
gr_indic	The faunal group
var_taxo	Taxonomical variable
ibgn_etat	Biological state associated with IBGN
ibgn_note	A note for IBGN
ibd	Biological diatom index
ibd2007	IBD measure before 2007
ibd_etat	Biological state associated with IBD
ibd_note	A note for IBD
ibd_ibgn	Pondered value including IDB and IBGN measures

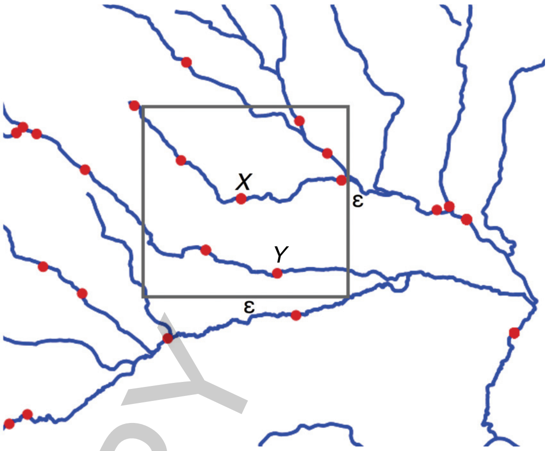


Fig. 6. Construction of zones using *Lambert coordinates*. (Colours are visible in the online version of the article; http://dx. doi.org/10.3233/IDA-160806)

6.1.3. Saône watershed dataset

Rapid population growth and human activities development (such as agriculture, industry, transports…) have increased vulnerability risk for water resources. In this context, river pollution is a phenomenon which is observed by measuring physicochemical and biological indicators for water quality. These indicators which evolve over time, depend explicitly on the location of sampling stations strategically located along several rivers.

The monitoring of stations placed along the Saône watershed produces several hydrological data. This data has been supplied by the RMC water agency[5] in the context of the *Fresqueau project*.[6] This project aims to develop new data mining techniques related to the water data management.

The data at our disposal is geo-referenced and temporally variable, thus making it difficult to explore globally. Moreover, the spatial relationship between the studied objects (i.e., monitoring stations) is implicit. It is therefore necessary to perform pre-processing that takes into account different spatial proximities (e.g., grouping stations according to their distance, according to their association to the same zone…) in order to build homogeneous zones and determine their neighboring areas.

In this work, our proposal consist on pre-processing data to bring together some monitoring stations and building homogeneous zones of spatial objects. For this purpose, the space is divided into areas grouping stations by exploiting their Lambert coordinates. In each of these zones, stations that are located within an area of size ϵ centered on station X are grouped, even if these stations belong to different watercourses. As can be clearly seen in Fig. 6, stations X and Y are considered to be in the same area, even if they are not on the same watercourse. For example, pesticide used in a crop field located between stations X and Y can impact on monitoring stations measures located on rivers around this crop field even if stations are not positioned in the same river. In our experimentations, ϵ value was set to 10 km.

Concerning the data itself, two types of data are available: information related to the monitoring station and information which corresponds to data measured by the station. Table 10 shows the attributes for Saône river dataset.

[5] http://www.eaurmc.fr/.

[6] http://engees-fresqueau.unistra.fr/.

Table 11
Characteristics of real datasets

Dataset	Number of zones	Number of dates by zone (average)	Number of items by date (average)	Time granularity	Spatial granularity
Dengue	12	23	22	Weekly	Quarters
Saône	223	17	5	Daily	100 km^2

Monitoring station information: each station is characterized by an identifier (id) and its spatial coordinates *(x, y)*. The Lambert Projection System 93 is used for the geo-referencing. A kilometric point is also provided and corresponds to the distance from the downstream confluence to the monitoring station following watercourse.

Monitoring station measurements: the stations measure biological indicators that determine the water quality of rivers. The frequency of these records varies with time and stations. Some stations have recurrent sample data while other stations only have a single sample data (e.g., for general monitoring). The main items associated with records are: (1) Standardized Global Biological Index (IBGN), a standardized measure based on identification of macro-invertebrates in rivers, and; (2) Biological Diatom Index (IBD), a standardized measure to diagnostic trophic pollutions (for more information, see e.g., [13]).

The IBGN and IBD measures have been processed based on: a note (e.g., *ibd_note*) and the current status of water quality (e.g., *ibgn_stat*). In addition, three other variables have been included in our dataset: (1) the taxonomic variety (*var_taxo*) representing the total number of taxa collected during a sampling, even if they are only represented by a single individual; (2) the faunal group that is the more sensitive to pollution (*gr_indic*), and; (3) the IBD measure established before the DCE regulation in France (*ibd2007*). All these information is used to estimate the condition of the watercourse at a specific survey point. Overall, the Saône dataset contains, in average, 17 dates by zone and 5 items by time.

More details of the characteristics of the real datasets are given in Table 11.

6.2. Experimental protocol

In this section, we focus on the performance evaluation of our proposals w.r.t. five questions:

1. Which one of the two proposed algorithms is the more efficient?
2. What is the impact of neighborhood relationship topology on the S2P extraction process?
3. What is the impact of the number of zones and the number of dates by sequence on the extraction process?
4. Is the spatiotemporal participation index an effective pruning measure?
5. What is the impact of data density on the S2P extraction process?

To answer the first question, we apply both approaches (pattern growth and Apriori based algorithms) on synthetic datasets containing 10 zones, 70 dates and 5 items.

To answer the second question, we study the impact of neighborhood topologies using two synthetic datasets containing 10 zones, 50 dates and 5 items in average using both type of neighborhood: (1) the graph neighborhood relationship and; (2) the grid neighborhood relationship (cf. Section 6.1.1).

To answer the third question, we also study the impact of the number of zones on the extraction process. For this, we vary the number of zones (10 and 20 zones by dataset) and we apply DFS-S2PMiner algorithm on these datasets. The impact of the number of dates per sequence on the extraction process also has been studied varying the number of dates per sequence (50 and 70 dates) on synthetic data.

Finally, to answer the two last questions, we have used two geo-referenced real datasets to evaluate the efficiency of the pruning measure and to evaluate the global performance (qualitatively and quantitatively) of our approaches.

Table 12
Examples of spatio-sequential patterns extracted from dengue monitoring database

Frequent spatio-sequential patterns	STPi

⟨(mean_wind:<=3.20 mean_temper:<=23.55)
(θ·[waste_container:<=39.00;
comunity_gather:<=20.00;
nb_cas_dengue:<=6.00;
ihre_index:<=24.55])⟩
	0.70

⟨(ihre_index:>34.82 nb_cas_dengue:<=6.00)⟩	0.60
⟨(θ·[precip:<=0.10; indoor_deposit:(2126.00;2692.50]] (nb_cas_dengue:<=6.00) (θ·ihre_index:<=24.55)⟩	0.60
⟨(θ·ihre_index:>34.82) (nb_cas_dengue:<=6.00 mean_temper:<=23.55) (θ·comunity_gather:<=20.00) (θ·[nb_cas_dengue:<=6.00; ihre_index:<=24.55])⟩	0.60
...	

Table 13
Examples of spatio-sequential patterns extracted from Saône river database

Frequent spatio-sequential patterns	STPi
⟨(var_taxo_31-40) (θ·[ibgn_16-20; ibgn_etat_very_good])⟩	0.32
⟨(ibgn_11-15) (θ·ibgn_etat_very_good) (θ·var_taxo_31-40)⟩	0.29
⟨(var_taxo_21-30) (θ·[ibgn_11-15; var_taxo_21-30; ibgn_etat_mean])⟩	0.25
...	

The two proposed algorithms were developed in Java. The experimentations have been performed on a PC based on Intel (R) Xeon (R) with 16 GB of RAM with Ubuntu Server 9.10 as operating system. The results are discussed in next section.

6.3. *Results*

In this section, we present a qualitative evaluation by giving some examples of extracted spatio-sequential patterns and comparing these patterns with other patterns found by the algorithm proposed by Tsoukatos et al. [23]. Then, we present a quantitative evaluation to estimate the performance of our approaches.

6.3.1. *Qualitative results*

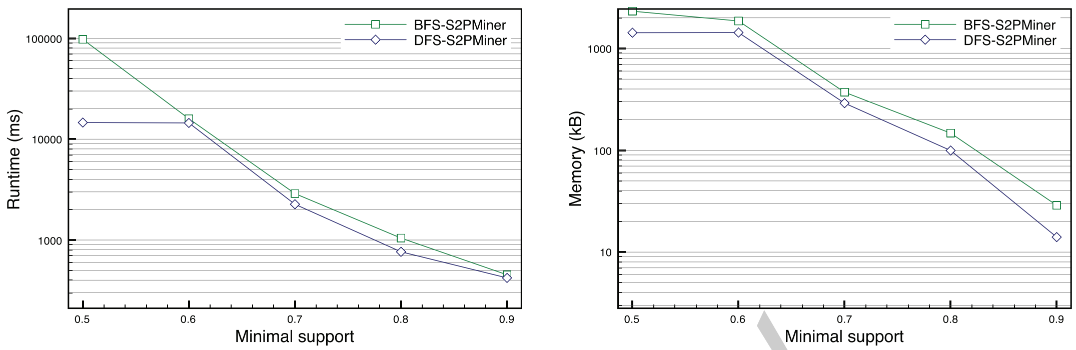
Tables 12 and 13 show spatial sequential patterns extracted during the execution of our algorithm applied to the dengue monitoring and Saône river databases respectively, using a minimum threshold of 0.6 and 0.1 respectively.

Considering the patterns obtained from dengue monitoring data set (see Table 12), we can see that they take into account the neighboring environment. For example, the second pattern in Table 12 can be interpreted by: at time t_0 we found a low presence of precipitations and house water deposits in two different neighboring areas, after, in studied area, we discover some cases of dengue, later, we detect the development of mosquitoes nest in a third neighboring area.

The results obtained in [23] do not contain this kinds of patterns. As we described in the related work, Tsoukatos et al. are looking for a sequence of events that occur frequently in a spatiotemporal database, without considering a dynamic neighborhood as we have shown in our examples (see Tables 12 and 13). Furthermore, quantitatively, our approach and Tsoukatos approach are not comparable due to the difference between the search space generated for both algorithms.

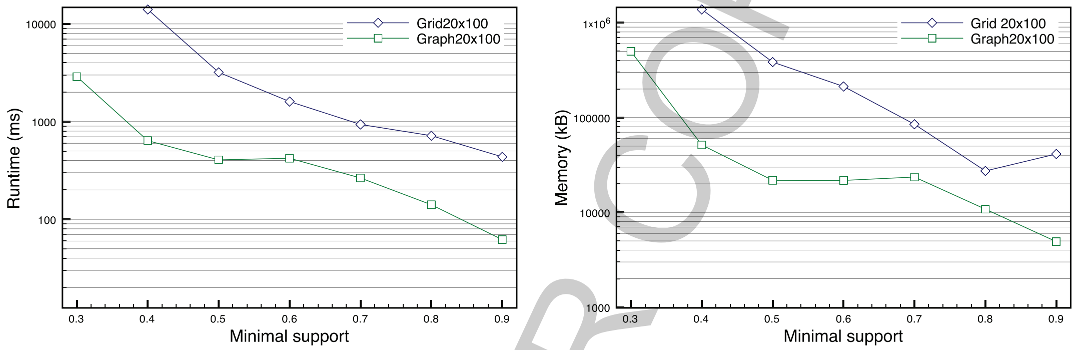
6.3.2. *Quantitative results*

To evaluate which of the two approaches (depth-first-search or level-wise) is more efficient, we apply both algorithms on synthetic datasets using graph neighborhood relationship (Graph20x70). Figure 7



(a) (b)

Fig. 7. Comparing the effectiveness of the two approaches (BFS-S2PMiner and DFS-S2PMiner) considering (a) runtime (b) memory. (Colours are visible in the online version of the article; http://dx.doi.org/10.3233/IDA-160806)



(a) (b)

Fig. 8. Evaluating the impact of different neighborhood relationship topology considering (a) runtime (b) memory. (Colours are visible in the online version of the article; http://dx.doi.org/10.3233/IDA-160806)

shows the runtime and the allocated memory for different thresholds. We can observe that the algorithm based on a depth-first-search approach is more efficient with respect to running time and memory use. This conclusion is endorsed by many works in the literature (see, e.g., [6]).

As detailed in Section 6.2, we study the impact of the neighborhood relationship topology on S2P extraction process for different thresholds using two synthetic dataset: Graph20x100 and Grid20x100. Figures 8(a) and (b) show the impact of both types of neighborhood relationship (grid based and graph based) on the execution time and the allocated memory respectively. In these figures, we observe that the execution time increases rapidly for the dataset generated using the grid based neighborhood relationship (8 neighbors by zone). In contrast, using the second neighborhood relationship, we get better results. These results show that the neighborhood relationship has an important impact in the extraction process, particularly on memory occupancy.

Thereafter, we want to estimate the impact of the number of zones and times per sequence on the pattern growth algorithm. Figure 9 shows the impact of zones and dates variation on the S2P extraction process respectively using four datasets: Graph10x70, Graph20x70, Graph10x50 and Graph10x70. It is clear that, for dataset containing 20 zones and 70 dates per sequence, the execution time and the memory occupancy is higher. We can also notice that, an increase of the number of dates increases the computational resources more than an increase of the number of zones.

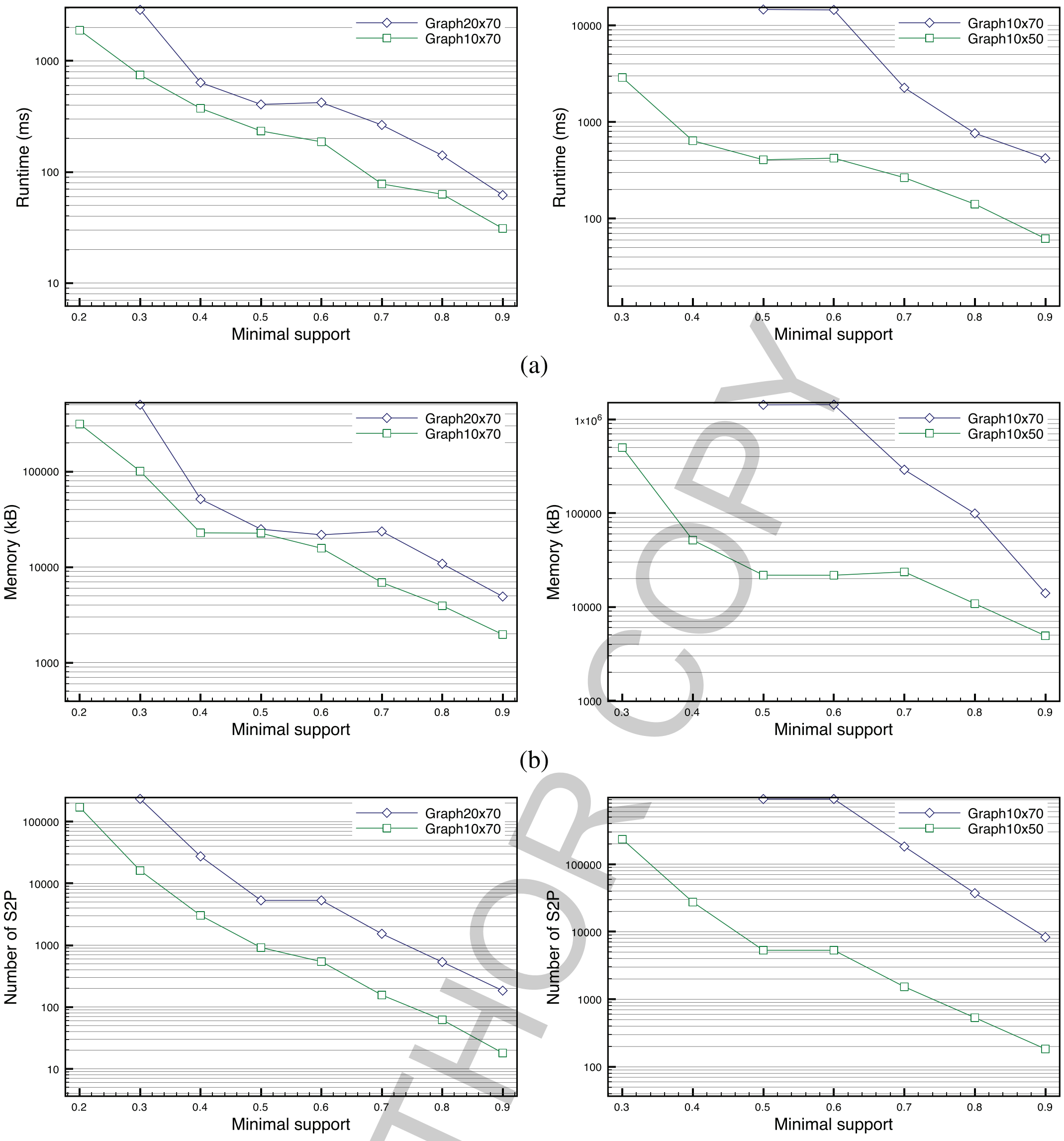
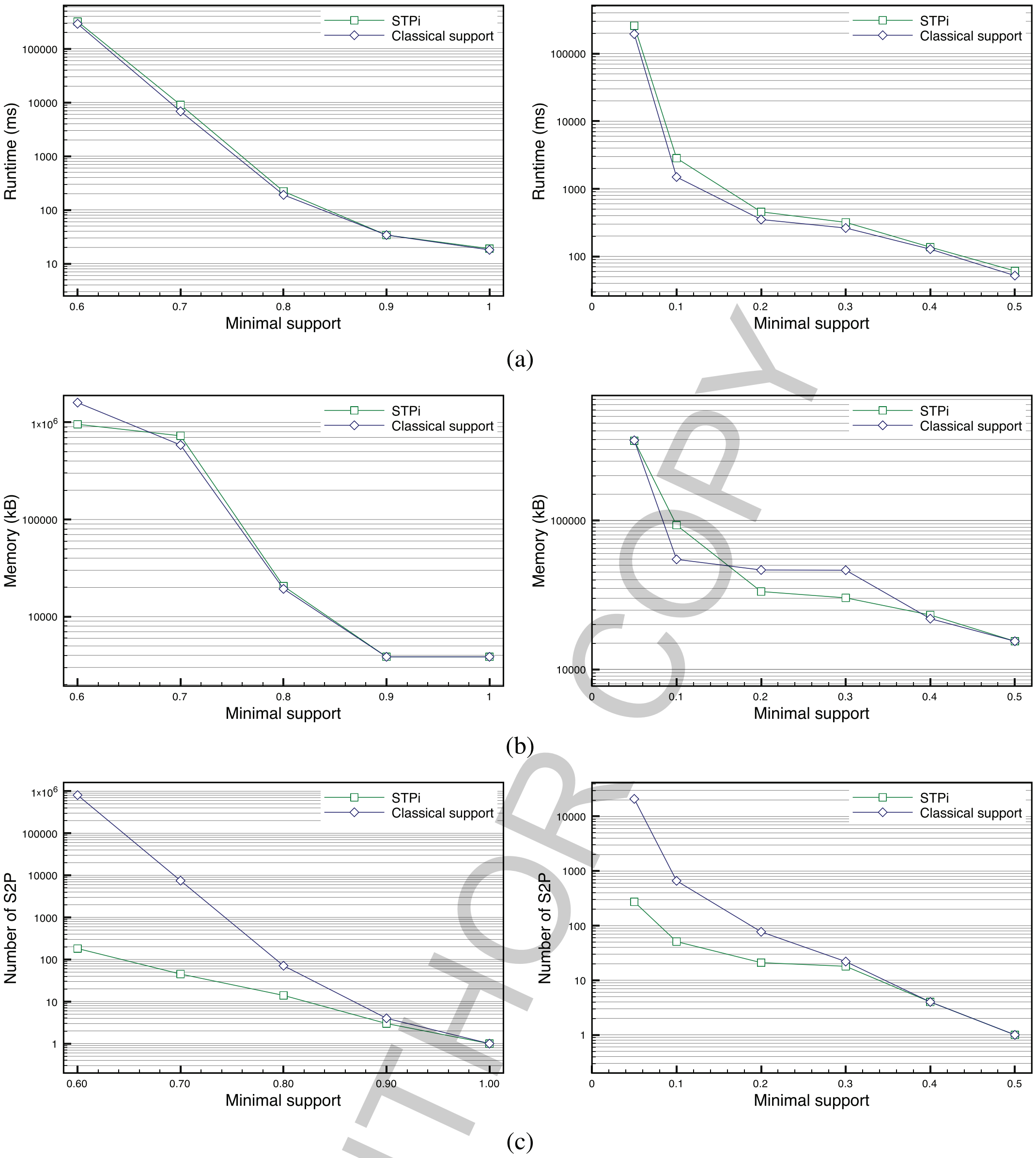


Fig. 9. Evaluating the impact of the variation of the number of zones and the number of dates on synthetic dataset considering (a) runtime (b) memory (c) number of S2P extracted. (Colours are visible in the online version of the article; http://dx.doi.org/10.3233/IDA-160806)

Finally, we evaluate the effectiveness of the spatiotemporal participation index (STPi) measure proposed in Section 4.2 and we show the global results. Purposely, we apply DFS-S2PMiner algorithm on two real datasets. In Fig. 10, the number of patterns extracted using the STPi measure is lower than the number of patterns extracted using the classical support. These results show that the STPi is very restrictive due to the temporal participation index (TPi). Indeed, this support filters the results by keeping only the patterns that appear several times in the spatiotemporal database. We see this phenomenon in the dengue datasets and more clearly on the river datasets (see Fig. 10). The results show the ease



(a)

(b)

(c)

Fig. 10. Evaluation of STPi measure effectiveness using Saône river and dengue datasets respectively on (a) runtime (b) memory (c) number of S2P extracted. (Colours are visible in the online version of the article; http://dx.doi.org/10.3233/IDA-160806)

of mining the watershed Saône dataset due to the time granularity and consequently the low density of data. For this dataset the extraction processes begins with a minimal threshold of 0.5 and we obtain good performances even using a low threshold.

We can notice that, for dense datasets (e.g., dengue monitoring dataset), the execution time and the memory occupancy is higher. In contrast, for datasets containing a significant number of null values, the execution time and the memory occupancy of S2P extraction process is lower (e.g., Saône river dataset).

DFS-S2PMiner algorithm has a good performance in datasets containing short sequences (e.g., Saône river dataset). Unfortunately, when mining long sequences (i.e., dense datasets) or when using very low threshold, the performance of our algorithm degrades dramatically.

Despite the reduced number of extracted patterns, the runtime and the allocated memory for *STPi* and *Support* remains the same for higher thresholds but differs for lower thresholds. This result can be explained by the fact that the process of computing the STPi measure increases the overall complexity of the extraction process. We have computed the mean square error between curves in Fig. 10(a) and (b) on the left side. These values are 172.15 sec. and 86265.5 kB, respectively for runtime and memory occupancy for dengue dataset.

Finally, the algorithm based on the depth-first-search strategy is more efficient that the algorithm based on the *Apriori* approach. This difference is due to the expensive candidate sequences generation used by the level-wise algorithms.

7. Conclusion and perspectives

In this paper, we propose a new concept of spatiotemporal patterns called spatio-sequential patterns *(S2P)*. This new type of patterns is used to analyze changes in a zone over time taking into account the neighboring environment. This concept describes, for example, the spatial and temporal evolution of dengue depending on the characteristics of zones (quarters) and their neighborhood. A formal framework is established to define the *S2P* generically.

To extract these patterns, we propose two generic algorithms called DFS-S2PMiner based on a depth-first strategy and BFS-S2PMiner based on a level-wise approach. We also proposed two interestingness measures which take into account spatial and temporal aspects. We tested our algorithm and the two defined measures on five synthetic datasets and two real datasets.

Among possible future developments of our work, we first try to mine only a compact set of spatiotemporal patterns using, for instance, the notion of top-k patterns or maximal patterns. Later, we want to filter the interesting patterns integrating a declarative patterns mining approach for spatiotemporal databases or integrating constraints in the patterns mining process. Moreover, it is important to analyze the patterns and to provide a visualization tool to help decision makers.

References

[1] R. Agrawal and R. Srikant, Fast algorithms for mining association rules in large databases, in: *Proceedings of the 20th International Conference on Very Large Data Bases*, VLDB '94, San Francisco, CA, USA, Morgan Kaufmann Publishers Inc. (1994), 487–499.

[2] R. Agrawal and R. Srikant, Mining sequential patterns, *Data Engineering, International Conference on* **0** (1995), 3.

[3] H. Alatrista-Salas, S. Bringay, F. Flouvat, N. Selmaoui-Folcher and M. Teisseire, The pattern next door: Towards spatio-sequential pattern discovery, in: *Proceedings of the 16th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD'12)*, (Jun 2012).

[4] H. Cao, N. Mamoulis and D. Cheung, Mining frequent spatio-temporal sequential patterns, *Proc of IEEE ICDM*, (2005), 82–89.

[5] M. Celik, S. Shekhar, J. Rogers and J. Shine, Mixed-drove spatiotemporal co-occurrence pattern mining, *Proc of IEEE TKDE* **20**(10) (2008), 1322–1335.

[6] C. Chand, A. Thakkar and A. Ganatra, Sequential pattern mining: Survey and current research challenges, *International Journal of Soft Computing and Engineering* **2**(1) (2012), 185–193.

[7] L. Geng and H.J. Hamilton, Interestingness measures for data mining: A survey, *ACM Computing Surveys (CSUR)* **38**(3) (Sept 2006).

[8] F. Giannotti, M. Nanni, F. Pinelli and D. Pedreschi, Trajectory pattern mining, *Proc of ACM SIGKDD*, (2007), 330–339.

[9] J. Han, K. Koperski and N. Stefanovic, Geominer: A system prototype for spatial data mining, in: *Proc of ACM SIGMOD*, SIGMOD '97, New York, NY, USA, ACM (1997), 553–556.

[10] J. Han, J. Pei, B. Mortazavi-Asl, Q. Chen, U. Dayal and M.-C. Hsu, Freespan: Frequent pattern-projected sequential pattern mining, in: *Proc of ACM SIGKDD*, KDD '00, New York, NY, USA, ACM (2000), 355–359.

[11] Y. Huang, L. Zhang and P. Zhang, A framework for mining sequential patterns from spatio-temporal event data sets, *Proc of IEEE TKDE* **20**(4) (2008), 433–448.

[12] M.V. Joshi, G. Karypisx and V. Kumar, A universal formulation of sequential patterns, in: *Proceedings of KDD (2001), Workshop on Temporal Data Mining* **1** (2001), 7.

[13] M. Lafont, A conceptual approach to the biomonitoring of freshwater: The ecological ambience system, in: *The Journal of Limnology* **60**(1) (2001), 17–24.

[14] L. Wang, K. Hu, T. Ku and X. Yan, Mining frequent trajectory pattern based on vague space partition, in: *Knowledge-Based Systems* **50** (2013), 100–111.

[15] Y.C. Liu, Discovering forward sequences from temporal data, in: *Knowledge-Based Systems* **39** (2013), 67–78.

[16] K. McGarry, A survey of interestingness measures for knowledge discovery, *The Knowledge Engineering Review* **20**(1) (Mar 2005), 39–61.

[17] P. Mohan, S. Shekhar, J.A. Shine and J.P. Rogers, Cascading spatio-temporal pattern discovery: A summary of results, in: *SDM* (2010), 327–338.

[18] B. Mortazavi-Asl, H. Pinto and U. Dayal, PrefixSpan: Mining sequential patterns efficiently by prefix-projected pattern growth, *Proc of 17th International Conference on Data Engineering* (2000), 215–224.

[19] J. Pei, J. Han, B. Mortazavi-Asl, J. Wang, H. Pinto, Q. Chen, U. Dayal and M.-C. Hsu, Mining sequential patterns by pattern-growth: The prefixspan approach, *Proc of IEEE TKDE* **16**(11) (2004), 1424–1440.

[20] S. Shekhar and Y. Huang, Discovering Spatial Co-Location Patterns A Summary Of Results, *Advances in Spatial and Temporal Databases* (2001), 236–256.

[21] Arthur A. Shaw and N.P. Gopalan, Finding longest frequent trajectory of dynamic objects using association approaches, *Intelligent Data Analysis*, (2014), 637–651.

[22] R. Srikant and R. Agrawal, *Advances in Database Technology EDBT'96*.

[23] I. Tsoukatos and D. Gunopulos, Efficient mining of spatiotemporal patterns, *Advances in Spatial and Temporal Databases*, (2001), 425–442.

[24] J. Wang, W. Hsu and M. Lee, Mining generalized spatio-temporal patterns, in: *Database Systems For Advanced Applications*, Springer (2005), 649–661.

[25] T. Wu, Y. Chen and J. Han, Re-examination of interestingness measures in pattern mining: a unified framework, *Data Mining and Knowledge Discovery* **21**(3) (Nov 2010), 371–397.

[26] M. Yuan, Knowledge toward discovery about geographic dynamics in spatiotemporal databases, in: *Geographic Data Mining and Knowledge Discovery, Second Edition*, H.J. Miller and J. Han, eds, 2009, pp. 347–365.

[27] M.J. Zaki, Spade: An efficient algorithm for mining frequent sequences, *Machine Learning* **42**(1/2) (2001), 31–60.