

# Procesamiento de consultas espacio-temporales complejas sobre diferentes escenarios

**María G. Dorzán, Edilma O. Gagliardi, Natalia J. Ortiz**

Facultad de Ciencias Físico, Matemáticas y Naturales,  
Departamento de Informática, Universidad Nacional de San Luis,  
San Luis, D5700HHW, Argentina  
{ mgdorzan, oli, njortiz }@unsl.edu.ar

y

**Gilberto A. Gutiérrez Retamal**

Facultad de Ciencias Empresariales,  
Departamento de Auditoría e Informática, Universidad del Bío-Bío  
Chillán, Chile  
ggutierr@ubiobio.cl

## RESUMEN

En nuestro trabajo de investigación nos dedicamos a las bases de datos que guardan objetos espaciales que cambian su tamaño y/o posición a través del tiempo, llamadas Bases de Datos Espacio-Temporales (BDET). Se requiere que una BDET sea capaz de representar modelos muy cercanos al mundo real, con todo el dinamismo que él implica, y administrar objetos que, básicamente, cambian su ubicación a través del tiempo. Un aspecto importante de los movimientos de los objetos, es el escenario en el que estos ocurren: (1) movimiento sin restricciones, (2) movimiento restringido y (3) movimiento restringido a redes. Sobre estos diversos escenarios se requiere realizar consultas de tipo espacio-temporal, típicas (timeslice, intervalo, eventos y trayectoria) y complejas (reunión y cercanía).

## INTRODUCCIÓN

En general, la mayoría de las bases de datos se conciben para describir algún aspecto del mundo sobre un período de tiempo dado. Cualquier valor que se almacene describe el pasado; es decir, que un hecho definido en un registro de una base de datos es una afirmación sobre el estado del mundo en un momento dado. Existen muchas aplicaciones que requieren manejar objetos espacio-temporales, es decir, objetos cuya posición espacial y/o forma cambia en distintos instantes de tiempo. Tales cambios deben ser manejados por un Sistema de Administración de Bases Datos que debe proveer, entre otros servicios, métodos de acceso para procesar en forma eficiente consultas cuyos predicados contemplan relaciones espacio-temporales. Actualmente, las Bases de Datos Espacio-Temporales (BDETs) en conjunto representan un tópico de estudio reciente y de mucho interés debido a la estrecha relación existente entre espacio y tiempo en una gran cantidad de aplicaciones. Como consecuencia, diariamente se produce una gran cantidad de datos espacio-temporales y por lo tanto, se necesita analizarlos y consultarlos eficientemente lo que conlleva el desarrollo de técnicas sofisticadas.

En investigaciones previas, nos hemos concentrado en responder varios tipos de consulta espacio-temporales como, timeslice, intervalo, eventos y trayectoria. En la actualidad nos abocamos a responder consultas más complejas debido a que las aplicaciones reales así lo demandan.

La organización del presente artículo es la siguiente: en la Sección 1, hablamos sobre los diferentes escenarios en dónde pueden ocurrir los movimientos de los objetos; en la Sección 2, describimos los distintos tipos de consultas que pueden ser resueltos en una BDET; en la Sección 3, nos referimos a los dos métodos de indexación desarrollados sobre los cuales implementamos tipos de consultas y sobre los cuales seguimos trabajando; y por último, en la Sección 4, mostramos el trabajo actual y futuro.

## 1. INDEXANDO OBJETOS EN MOVIMIENTO

Movilidad es una palabra clave que se tiene en cuenta debido a todas aplicaciones y servicios ocurrentes en la realidad y, en particular, con la llegada de nuevas tecnologías, como las de computación móvil. Estas aplicaciones necesitan de una disponibilidad de información que sólo puede ser provista por una BDET, capaz de almacenar y recuperar los datos de manera eficiente [10]. Para ello, se requieren de métodos de indexación que consideren tanto el almacenamiento eficiente, sobre todo de los atributos dinámicos del objeto móvil, más una actualización constante de esos atributos, y de una respuesta adecuada frente a los diversos tipos de consultas.

Un aspecto importante de los movimientos de los objetos, es el escenario en el que ocurren. Para categorizar varios enfoques de indexación se han usado tres escenarios diferentes: *movimiento sin restricciones*, *movimiento restringido* y *movimiento restringido a redes*. A continuación describimos dichos escenarios.

### MOVIMIENTO SIN RESTRICCIONES

El movimiento sin restricciones es aquel en donde los objetos se mueven libremente dentro de la región en la que se encuentran. Ejemplos de este contexto son, entre otros, el monitoreo de animales que luego de permanecer en cautiverio, son reubicados en sus hábitats naturales. Dentro de los movimientos sin restricciones, existen distintos métodos de indexación que se diferencian de acuerdo a la información que indexan: métodos que indexan el pasado; métodos que indexan el presente y métodos que indexan el presente y futuro [9].

### MOVIMIENTO RESTRINGIDO

El movimiento restringido es aquel en que el escenario se presenta limitado para los objetos que se mueven. Esta limitación tiene que ver obstáculos (*infraestructuras*) que obligan al objeto a tomar caminos alternativos ya que éstos no pueden ser sorteados. Un ejemplo de este contexto es el de un robot autónomo, al cual se le indica qué hacer, pero no cómo hacerlo, es decir, planifica sus movimientos en base al conocimiento que tiene de un determinado ambiente y sus obstáculos [10].

### MOVIMIENTO RESTRINGIDO A REDES

En muchas aplicaciones de la realidad, los movimientos se modelan basándose en una red más que en un espacio bidimensional. Resulta interesante pensar en un movimiento restringido a una red, porque no interesa el espacio subyacente a ella para moverse, es decir, al ver una carretera y un automóvil moviéndose en ella, no interesa el espacio o el tamaño de banquina adyacente a la misma, como así tampoco la posición del objeto en términos de coordenadas  $x$  e  $y$ , sino que interesa la posición relativa respecto de la red, es decir, en el kilómetro 24 de la ruta 3, por ejemplo [10]. A menudo, una red dirige el tráfico de los objetos que se mueven en ella. Es decir, al tener una red predefinida, los objetos que se mueven en ella tienden a moverse siguiendo los diferentes caminos que la misma presenta. La red puede ser tanto artificial, creada basándose en un criterio para el movimiento de los objetos, como real, basada en una del mundo real [2].

## 2. CONSULTANDO OBJETOS EN MOVIMIENTO

Todo método de acceso debe proveer la capacidad de poder responder eficientemente consultas que involucran predicados espacio-temporales, reflejando lo que es de interés para el usuario. Además es quien determina de qué manera tratar y evaluar la consulta. Al momento de utilizar un índice, es importante determinar qué tipo de consultas se tendrán en cuenta ya que esto establece la forma en que se administra y almacena la información, como así también la manera de recuperar la información necesaria.

Los tipos de consulta espacio-temporales tratados en la literatura afín son los siguientes:

- *Instante de tiempo* o *TimeSlice*: consiste en, dado un instante de tiempo  $t_i$ , y una ventana sobre el plano, informar todos los objetos que están o pasaron por ella durante ese instante de tiempo [8].
- *Intervalo de tiempo*: consiste en, dados dos instantes de tiempo  $t_i$  y  $t_j$ , y una ventana sobre el plano, informar todos los objetos que están o pasaron por ella durante ese intervalo de tiempo [8].
- *Eventos*: dado un instante de tiempo  $t_i$ , y una ventana sobre el plano, informar los objetos que aparecieron/desaparecieron en ella durante ese instante de tiempo [8].
- *Trayectoria*: dado un objeto, y un intervalo de tiempo, informar las posiciones que ocupó dicho objeto durante ese intervalo de tiempo [8].
- *Ensamble (Join)*: Este tipo de consulta consiste en, dados dos conjuntos de datos, recuperar todos los pares de objetos que satisfacen un predicado específico del tipo de dato que se trate (espacial, temporal, espacio-temporal). Existe una amplia variedad de predicados; sin embargo, según [11], el predicado de intersección tiene un rol muy importante ya que a partir de él se pueden obtener los restantes. Un ejemplo puede ser, “hallar todos los pares de objetos que se hallan espacialmente cerca, es decir, dentro de una distancia  $D$ , durante un intervalo o instante de tiempo específico”. Otra aplicación inmediata es la detección de accidentes comparando la trayectoria de los vehículos: “hallar los pares de vehículos que estarán cerca, en una radio de 10 Km., dentro de 5 minutos”.
- *Cercanía o proximidad*:
  - *k-vecinos más cercanos*: Dado un objeto  $X$ , encontrar los  $k$  objetos  $X'$  que tienen la mínima distancia de  $X$ . Aquí, la distancia se define entre atributos espaciales como la distancia (Euclidiana o de Manhattan) entre sus puntos más cercanos. Un ejemplo para este tipo de consulta puede ser: “hallar las 5 ambulancias más cercanas, con respecto al lugar del accidente”; y uno más complejo sería: “hallar las 5 ambulancias más cercanas con respecto al lugar del accidente en un intervalo de tiempo de 2 minutos antes y después del accidente, sabiendo las direcciones y velocidades de las ambulancias y el mapa de las calles” [8]. Existen cuatro escenarios posibles teniendo en cuenta si el objeto  $X$  y el conjunto  $S$  de objetos de la base de datos se mueven o no: (1) no se mueve ni el objeto  $X$  ni los objetos de  $S$ ; (2) se mueven los objetos del conjunto  $S$  pero no el objeto  $X$ ; (3) se mueve el objeto  $X$  pero no los del conjunto  $S$ ; y (4) tanto el objeto  $X$  como los del conjunto  $S$  se mueven.
  - *El par de vecinos más cercano*: Dado un conjunto de objetos  $S$ , encontrar dos objetos cuya distancia mutua sea mínima. Un ejemplo típico en el control de tráfico aéreo sería “encontrar el par de aviones más cercano” ya que se determinaría dónde existe peligro de colisión.
  - *Patrones espacio-temporales*: Un patrón espacio-temporal se representa como una secuencia de distintos predicados espacio-temporales, donde interesa el ordenamiento temporal (exacto o relativo) de los predicados. Este tipo de consulta consiste en, dada una gran colección de trayectorias espacio-temporales y un patrón, recuperar todos los objetos que siguen ese patrón de movimiento definido en tiempo y espacio [7]. Un ejemplo sería, “identificar todos los vehículos que estuvieron muy cercanos a los tres ataques de francotiradores en Maryland (las posiciones y tiempos de los ataques son conocidos)” o “localizar los productos que abandonaron la fábrica hace un mes, fueron almacenados en uno de los almacenes cerca del muelle y fueron cargados en un barco”. Este tipo de consulta puede tener tipos arbitrarios de predicados espaciales (es decir, búsqueda por rango, vecino más cercano, etc.), donde cada predicado puede ser asociado con: (1) una restricción exacta temporal que es o un instante o un intervalo de tiempo (consulta de patrones

espacio-temporales con tiempo), o (2) generalmente, con un orden relativo (consulta de patrones espacio-temporales con orden).

### 3. MÉTODOS DE INDEXACIÓN TRATADOS

Hemos diseñado e implementado dos métodos de acceso espacio-temporal, cada uno basado en un escenario diferente. Hasta el momento, cada método responde eficientemente los primeros cuatro tipos de consulta nombradas en la sección anterior.

A continuación se describen brevemente ambos métodos, donde el escenario del primero es sin restricciones y el del segundo es restringido a redes.

- *D\*R-Tree* [4] es un método de acceso espacio-temporal que se basa en la idea expuesta en [5]. Mantiene puntos de referencia para ciertos instantes de tiempo en la base de datos, tal que para esos instantes los objetos se almacenan en una estructura de datos R-Tree. Las modificaciones ocurridas entre los puntos de referencia de tiempo consecutivos se mantienen en una estructura llamada Bitácora.

El método pretende mantener un equilibrio entre el espacio de disco utilizado por la estructura y el tiempo de acceso empleado en responder los distintos tipos de consulta. Nuestra propuesta tiene las siguientes características: (1) se utiliza un Índice para almacenar los instantes de tiempo que se establezcan como puntos de referencia; (2) Los R-Tree's son utilizados para almacenar la ubicación espacial de los objetos y cada R-Tree está asociado a un punto de referencia de tiempo, según el índice descrito en el punto anterior; (3) se usan Bitácoras para almacenar los movimientos realizados en instantes intermedios entre puntos de referencia almacenados en el índice de instantes; además, se guardan las referencias a los movimientos anteriores inmediatos por cada objeto; (4) se agrega un Índice para acceder a los últimos registros de movimiento en cada bitácora por cada objeto. Con ello se posibilita la entrada directa a las bitácoras involucradas en la trayectoria de los objetos.

- *I+MON-Tree* [3] es un método que extiende las capacidades del índice mencionado en [1], incorporando al almacenamiento de la información del pasado, información sobre posiciones actuales de los objetos y amplía el rango de consultas por ésta soportada, agregando las consultas timeslice y por trayectoria.

La estructura tiene las siguientes características: (1) incorpora un Índice de Información Actual que almacena cubos abiertos, que hacen referencia a objetos cuyo instante final en una posición aún no está definido. También se mantiene información que hace referencia a los cubos anteriores y que nos permite recuperar la trayectoria del objeto; (2) usa un 2D R-Tree [6] para indexar la mínima caja que recubre cada poligonal; (3) la estructura de hashing almacena el identificador de la poligonal y el puntero al 2D R-Tree [6] del nivel inferior que le corresponde a la misma; (4) se utiliza un conjunto de 2D R-Tree [6], en el nivel inferior, que indexan el movimiento de los objetos a lo largo de la poligonal. Cada movimiento de un objeto, indexado por éstos, mantiene información del movimiento anterior y/o posterior que conforma la trayectoria de un objeto.

### 4. TRABAJO ACTUAL Y VISIÓN DE FUTURO

En el transcurso de esta investigación, observamos que la mayoría de los métodos se abocan a responder eficientemente ciertos tipos de consultas, lo que nos motivó a desarrollar índices capaces de mantener la información necesaria para poder responder los cuatro tipos de consulta más solicitados en general: timeslice, intervalo, eventos y trayectoria. De esta forma, surgieron los dos métodos sobre dos escenarios totalmente diferentes, que permiten almacenar datos espacio-

temporales y responder eficientemente tipos de consultas sobre ellos, habiendo obtenido resultados satisfactorios en las evaluaciones experimentales.

Actualmente, proponemos ampliar las capacidades de nuestros índices para resolver los tipos de consultas complejas antes mencionados, join espacio-temporal y búsqueda de los vecinos más cercanos (sobre los distintos escenarios planteados), entre otras, dedicándonos al estudio y análisis del diseño de estructuras de datos y algoritmos que permitan resolver dichas consultas eficientemente. Consideramos que poder responder estos tipos de consulta resulta de un gran aporte, ya que, al implementarlos, se vincularía el área de BDET con la Geometría Computacional, introduciendo conceptos como cierre convexo, algoritmos de proximidad, entre otros.

Estos trabajos están enmarcados dentro de la Línea de investigación Geometría Computacional y Bases de Datos Espacio-Temporales, perteneciente al Proyecto Tecnologías Avanzadas de Bases de Datos 22/F314, Departamento de Informática, Universidad Nacional de San Luis; en el Proyecto AL07-PAC-027 Geometría Computacional, subvencionado por la Universidad Politécnica de Madrid; y en el marco de la Red Iberoamericana de Tecnologías del Software (RITOS2), subvencionado por CYTED. Por todo ello, se ha establecido un grupo de interés en el tema conformado por docentes investigadores de la Universidad Nacional de San Luis.

## REFERENCIAS BIBLIOGRÁFICAS

- [1] Almeida, V. y Güting, R. *Indexing the trajectories of moving objects in networks*. In Proc. of the 16th Intl. Conf. on Scientific and Statistical Database Management (SSDBM). (2004)
- [2] Brinkhoff, T. *Generating Network-Based Moving Objects*. In Proc. of the 12th Intl. Conference on Scientific and Statistical Database Management Berlin, IEEE Computer Society Press. (2000)
- [3] Correa, L. y Ortiz, N. *I+MON-Tree: índice espacio-temporal para objetos en movimiento*. Informe Trabajo Final Licenciatura, UNSL. (2006)
- [4] Dorzán M., Gagliardi E., Gómez Barroso J., y Gutiérrez Retamal G. *Un nuevo índice eficiente para resolver diversas consultas espacio-temporales*. Conferencia Latinoamericana de Informática. (2006)
- [5] Gutiérrez G. *A Spatiotemporal Access Method based on Snapshots and Events*. ACM GIS'05, Bremen, Germany. (2005)
- [6] Guttman, A. *R-Trees: A dynamic index structure for spatial searching*. In ACM SIGMOD Conf. on Management of Data, pages 47-57, Boston, ACM. (1984)
- [7] Hadjieleftheriou M., Kollios G., Bakalov P. y Tsotras V. *Complex Spatio-Temporal Pattern Queries*. (2005)
- [8] Manolopoulos, Y., Papadopoulos, A., Sellis Timos, K. y Theodoridis, Y. *Specifications for efficient indexing in spatiotemporal databases*. In Proc. 10th International Conference on Scientific and Statistical Database Management, Capri, Italy, pages 123-132. (1998)
- [9] Mokbel M., Ghanem T., y Aref W. *Spatio-temporal Access Methods*. IEEE Data Engineering Bulletin 26, pp. 40-49. (2003)
- [10] Pfoser, D. *Indexing the Trajectories of Moving Objects*. Bulletin of the IEEE Computer Society Technical Committee on Data Engineering. (2002)
- [11] Volker Gaede and Oliver Günther. *Multidimensional access methods*. ACM Computing Surveys, 30(2):170–231, (1998)