# Análisis de desempeño de algoritmos de detección de patrones de agrupamiento en Bases de Datos espacio-temporales

Sandra Cristina Muñoz Castillo* †, Nombre Otro Autor* †. *Universidad de Nariño. †Grupo de Investigación Aplicada en Sistemas (GRIAS).

**Resumen**—En esta parte se escribe el abstract en Español. Ell sid djd did djd d did didi did did didid didid diid diid didi did

**Abstract**—Here write abstract in English. did didid didid didid didid didid didid did didi did

**Keywords**—Proyecto, escribir, latex, otras palabras, importante.

## 1 Introducción

Los recientes avances tecnológicos y el amplio uso de localización por sistemas de posicionamiento global (GPS), identificación por radio frecuencia (RFID) y tecnologías en dispositivos móviles han hecho que las bases de datos espacio temporales recolectadas hayan incrementado con un porcentaje acelerado. Esta gran cantidad de información ha motivado a desarrollar eficientes técnicas, para procesar consultas acerca del comportamiento de los objetos en movimiento, como descubrir patrones de comportamiento entre las trayectorias de objetos en un periodo continuo de tiempo.

Los métodos que existen para consulta de trayectorias se centran principalmente en responder un único rango simple de predicado y consultas de vecinos mas cercanos por ejemplo: "encontrar todos los objetos en movimiento que se encontraban en la zona A a las 10 de la mañana (en el pasado)" o "encontrar el coche que condujo tan cerca como sea posible a la ubicación B durante el intervalo de tiempo de 10 de la mañana a 1 de la tarde". Recientemente, diversos estudios se han centrado en la consulta de los patrones para la captura del comportamiento de los objetos en movimiento reflejada en colaboraciones tales como clusters móviles [1] [2], consulta de convoyes [3] y patrones de agrupación [4] [5] [6]. Estos patrones descubren grupos de objetos en movimiento que tienen una "fuerte" relación en el espacio durante un tiempo determinado. La diferencia entre todos esos patrones es la forma de definir la relación entre los objetos en movimiento y su duración en el tiempo.

En este artículo se enfocará en el descubrimiento de patrones de agrupación, conocidos como "flocks", entre los objetos en movimiento de acuerdo a las características de los objetos de estudio (animales, peatones, vehículos o fenómenos naturales), como interactúan entre si y como se mueven juntos [7] [8]. [6] define patrones de agrupamiento como el problema de identificar todos los grupos de trayectorias que permanecen "juntas" por la duración de un intervalo de tiempo dado. Consideramos que los objetos en movimiento están suficientemente cerca si existe un disco con un radio dado que cubre todos los objetos que se mueven en el patrón (Figura 1). Una trayectoria satisface el patrón anterior, siempre y cuando suficientes trayectorias están contenidos dentro del disco para el intervalo de tiempo especificado, es decir, la respuesta se basa no sólo en el comportamiento de una trayectoria dada, sino también en los más cercanos a él. El enfoque actual para descubrir patrones de agrupación de movimiento consiste en encontrar un conjunto adecuado de discos en cada instante de tiempo y

- *S. Muñoz. Ingeniera de Sistemas and Magíster en Docencia Universitaia, Universidad de Nariño (Pasto - Colombia). E-mail: sandramunoz@udenar.edu.co.*
- *N. Apellido. Ingeniero. E-mail: mail@udenar.edu.co.*

*Artículo enviado 7 de abril, 2014.*

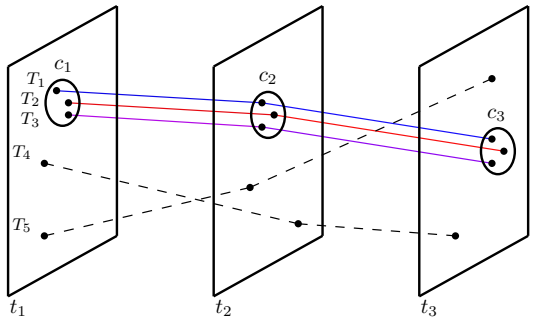


Figura 1. Ejemplo de patrones de agrupamiento

luego la fusión de los resultados de un instante de tiempo a otro. Como consecuencia, el rendimiento y el número de patrones final depende del número de los discos y cómo estos se combinan.

En el ejemplo de la Figura 1 se muestra un patrón de agrupamiento el cual contienen tres trayectorias T1, T2, T3 que están dentro de un disco en tres instantes de tiempo consecutivos. Los discos se pueden mover libremente en el espacio bidimensional con el fin de acomodar los tres objetos en movimiento y su centro no tiene por que ser cualquiera de las localizaciones de los objetos en movimiento. Esto hace que el descubrimiento de agrupación de patrones sea mucho más complicada porque hay un número infinito de posibles colocaciones del disco en cualquier instante de tiempo y el posible número de combinaciones convierten al problema en un problema NP-complejo.

La implementación de este algoritmo tiene diversas aplicaciones tales como: sistemas integrados de transporte, seguridad y monitoreo, seguimiento a grupos de animales y fenómenos naturales, encontrando así una alternativa diferente para solucionar los problemas en el mundo, analizando como se mueven los objetos en la tierra y cuales son los patrones de comportamiento que existen entre si.

En este artículo se muestra una comparación entre dos algoritmos, el propuesto por [6] y [9], con el fin de identificar los problemas asociados a su rendimiento, y probando estos algoritmos en distintos conjuntos de datos.

## 2 Trabajos Relacionados

Las capacidad de recolectar datos de objetos en movimiento ha ido aumentando rápidamente y el interés de consulta de patrones que describen el comportamiento colectivo también ha aumentado. [6] enumera tres grupos de patrones “colectivos” en bases de datos de objetos en movimiento: clústers móviles, consulta de convoyes y patrones de agrupación.

Los clústers móviles [1] [2] [10] y consultas de convoyes [3] [11], tienen en común que se basan en algoritmos de clústering, principalmente en algoritmos basados en densidad como el algoritmo DBSCAN [12].

Los clústers móviles se definen entre dos instantes de tiempo consecutivos. Los clústers se pueden unir sólo si el número de objetos comunes entre ellos están por encima del parámetro predefinido. Un clúster es reportado si no hay otro nuevo clúster que pueda ser unido a este. Este proceso se aplica cada vez para todos los instantes de tiempo en el conjunto de datos.

Las consultas de convoyes se definen como un clúster denso de trayectorias que permanecen juntas al menos por un por un tiempo continuo predefinido.

Las principales diferencias entre las dos técnicas son la forma en que se unen los grupos entre dos intervalos consecutivos de tiempo y el uso de un parámetro adicional para especificar un tiempo mínimo de duración. Aunque estos métodos están estrechamente relacionados con los patrones de agrupamiento, ninguno de ellos asume una forma predefinida.

Previos trabajos de detección de patrones de agrupamiento móviles son descritos por [4] y [5]. Ellos introducen el uso de discos con un radio predefinido para identificar grupos de trayectorias que se mueven juntos en la misma dirección, todas las trayectorias que se encuentran dentro del disco en un instante de tiempo particular se considera un patrón candidato. La principal limitación de este proceso es que hay un número infinito de posibles ubicaciones del disco en cualquier instante de tiempo. En efecto, [4] han demostrado que el descubrimiento de agrupaciones fijas, donde los patrones de las mismas entidades permanecen juntas durante todo el intervalo, es un problema NP-complejo.

[6] on los primeros en presentar una solución exacta para reportar patrones de agrupación en tiempo polinomial, y también pueden trabajar efectivamente en tiempo real. Su trabajo revela

que el tiempo de solución polinomial se puede encontrar a través de la identificación de un número discreto de ubicaciones para colocar el centro del disco. Los autores proponen el altoritmo BFE (Basic Flock Evaluation) basado en el tiempo de unión y combinación de los discos. La idea principal de este algoritmo es primero encontrar el número de discos válidos en cada instante de tiempo y luego combinarlos uno a uno entre tiempos adyacentes. Adicionalmente se proponen otros cuatro algoritmos basados en métodos heurísticos, para reducir el número total de candidatos a ser combinados y, por lo tanto, el costo global computacional del algoritmo BFE. Sin embargo, el pseudocódigo y los resultados experimentales muestran todavía una alta complejidad computacional, largos tiempos de respuesta y un gran número de patrones que hace difícil su interpretación.

[9] propone una metodología que permite identificar patrones de agrupamiento en movimiento utilizando tradicionales y potentes algoritmos de minería de datos usando reglas de asociación, el cual fue comparado con BFE demostrando un alto rendimiento con conjuntos de datos sintéticos, aunque con conjuntos de datos reales el tiempo de respuesta siguió siendo eficiente pero similar a BFE. Este algoritmo trata el conjunto de trayectorios como una base de datos transaccional al convertir cada trayectoria, que se define como un conjunto de lugares visitados, en una transacción, definida como un conjunto ítems. De esta manera, es posible aplicar cualquier algoritmo de reglas de asociación y encontrar patrones frecuentes sobre el conjunto dado.

# 3 Implementación

Se implementaron los algorítmos BFE y LCMFLOCK basados en el pseudo-código publicado por [6] y [9] respectivamente, usando python version 3.

## 3.1 BFE

Este algoritmo se divide en dos partes: la primera parte, encontrar los discos que contienen el mayor número de puntos dado un radio ($\epsilon$) y un número mínimo de puntos ($\mu$) dentro de ese radio para instante de tiempo. La segunda parte: encontrar el número de puntos que se mueven juntos (flocks) en un tiempo mímimo ($\delta$).

Para la primera parte es necesarios la utilización tanto de diccionarios de datos como estructuras kd-tree para la búsqueda del vecino más cercano, en esta implementación se uso la clase scipy.spatial.cKDTree de SciPy [1] la cual proporciona un índice dentro de un conjunto de puntos k-dimensionales que se pueden utilizar para buscar rápidamente los vecinos más cercanos de cualquier punto.

## 3.2 LCMFLOCK

Este algoritmo usa la primera parte del algoritmo de BFE que es encontrar los discos maximos, en la segunda parte para abordar el problema de combinatoria utiliza un enfonque de patron frecuente de mineria, en el cual se contruyo un diccionario de datos asociando la localizacion de los puntos de cada trayectoria con su respectivo disco en orden para generar una version transaccional del conjunto de datos. Este conjunto es pasado como parametro junto con el umbral mínimo de soporte (min_sup), para el algoritmo de LCM, disponible para descargar en [13], están disponibles dos variantes del programa; LCM_max y LCM_closed los cuales recuperarán el conjunto máximo o cerrado de los patrones de frecuencia, dependiendo del caso. La salida M es un archivo de texto sin formato, donde cada línea es un patrón máxima que contiene un conjunto de ID's de discos separados por espacios.

# 4 Experimentación Computacional

Los resultados fueron producidos usando conjuntos de datos sintéticos y reales en una máqina Dell OPTIPLEX 7010 con procesador Intel®Core™i7-3770 CPU de 3.40GHz x 8, 16 GB de RAM y 1TB 7200 RPM de Disco Duro, corriendo Ubuntu con linux 3.5. Para todos los casos se usaron los algoritmos implementados en python version 3.

1. Scipy es un ecosistema basado en Python, software de código abierto para las matemáticas, la ciencia y la ingeniería. http://www.scipy.org/

TABLA 1
Conjunto de datos sintéticos

| Dataset | Network | Number of Trajectories | Number of Points | Trajectory size (avg) |
|---|---|---|---|---|
| SJ25KT60 | San Joaquin | 25000 | 992140 | 40 |
| SJ50KT55 | San Joaquin | 50000 | 2014346 | 37 |
| TAPAS Cologne | Cologne, Germany | 49225 | 1813454 | 37 |
| Original_Beijing | Beijing, China | 23800 | 1207110 | 50 |
| Alternative_Beijing | Beijing, China | 18216 | 760814 | 42 |

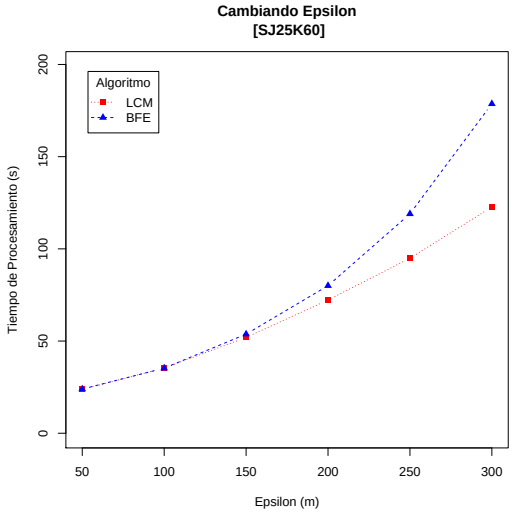


Figura 2. Caso de Prueba: SJ25K60

## 4.1 San Joaquin

Un grupo de conjuntos de datos sintéticos fueron creados usando un modelo para la generación de objetos en movimiento, como se describe en [14]. Dos conjuntos de datos sintéticos fueron creados usando la red de San Joaquín proporcionada en el sitio web del generador basado en red [15]. El primer conjunto de datos recoge 992140 lugares simulados para 25.000 objetos en movimiento durante 60 instantes de tiempo. El segundo recoge 50.000 trayectorias de 2.014.346 de puntos durante 55 instantes de tiempo. La Table 1 resume la información principal. Es importante aclarar que el tamaño de la trayectoria se refiere al número promedio de ubicaciones de puntos e intervalos de tiempo en lugar de a la longitud espacial media.

## 4.2 TAPAS Cologne

Este conjunto de datos sintético se preparó utilizando el escenario TAPAS Cologne [16] en SUMO [17], un reconocido simulador de tráfico para la movilidad urbana. El escenario de simulación TAPAS Colonia describe el tráfico dentro

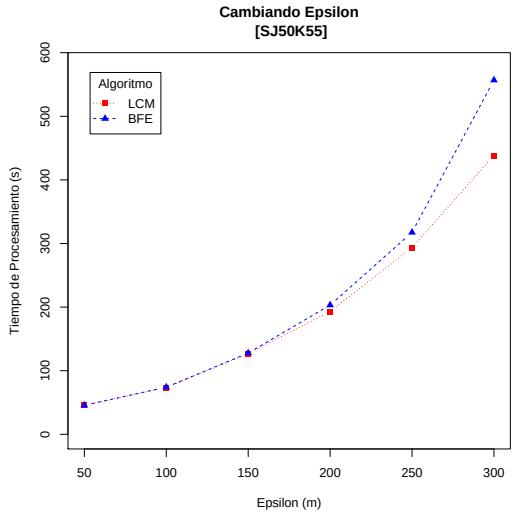


Figura 3. Caso de Prueba: SJ50K55

de la ciudad de Cologne (Alemania) durante un día entero. La principal ventaja de este conjunto de datos es que sus trayectorias no se generan aleatoriamente. Los datos de la demanda original, se deriva de TAPAS, un sistema que calcula el deseo de movilidad para una población de la zona generada con base en la información sobre los hábitos de viaje de los alemanes y en la información sobre la infraestructura de la zona en que viven [18]. El conjunto de datos original es enorme por lo que sólo una esta disponible al público la versión de 2 horas [19]. Debido a la memoria constriñe las trayectorias más cortas que se podaron 20 minutos. El último conjunto de datos recoge 49.225 trayectorias y más de 1,8 millones de puntos. La Tabla 1 describe los detalles sobre el conjunto de datos.

## 4.3 Movimiento de peatones en Beijing

Este conjunto de datos reales recopila información de movimiento de un grupo de personas en todo el área metropolitana de Beijing, China, el uso de un conjunto de datos de la trayectoria GPS proporcionado por [20]. El conjunto de datos se recogieron durante el proyecto Geolife por 165 usuarios anónimos en un período de dos años entre abril de 2007 y agosto de 2009. Ubicaciones eran grabada por diferentes registradores GPS o Smartphones y la mayoría de ellos presentan una frecuencia de muestreo alta. La región alrededor de la “5th Ring Road” en el área metropolitana de Beijing mostró la mayor concentración

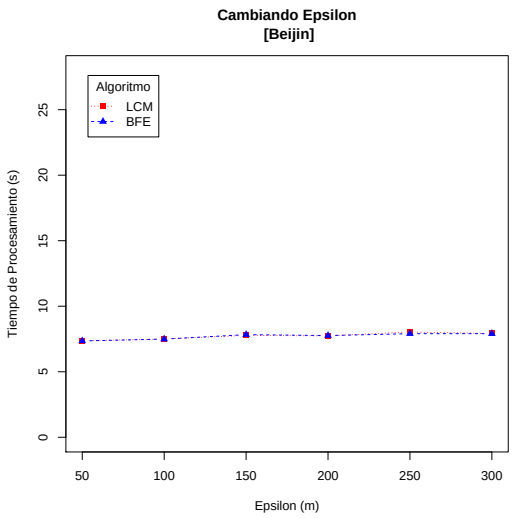


Figura 4. Caso de Prueba: Beijin

de trayectorias. Esto fue usado para generar un conjunto de datos de muestra. Cada trayectoria fue interpolada por minuto (un punto por minuto) y saltos de 20 minutos o más sin señal se utilizaron para marcar una nueva trayectoria. Por último, el conjunto de datos recoge más de 1,2 millones de ubicaciones de puntos y 23.800 trayectorias. Sin embargo, como este conjunto de datos seguimiento poca cantidad de entidades en movimiento (165 usuarios) en una ventana de tiempo (más de 2 años) no hay mucho trayectorias fueron sucediendo al mismo tiempo. Para probar la escalabilidad se decidió crear un conjunto de datos alternativo basado en las trayectorias reales, pero todos ellos a partir de la mismo tiempo. Una vez más, para la memoria limita las trayectorias más cortas que 10 minutos y más de 3 horas se podaron. El conjunto de datos alternativa almacena 760.814 ubicaciones de los puntos y 18.216 trayectorias reales que ocurren juntos. La Tabla 1 resume los detalles para ambos conjuntos de datos.

## 5 Conclusiónes

Aquí las conclusiones

## Agradecimientos

Aquí se agradece a la facultad y al programa y la universidad ... si es investigación a quien la financio

## Referencias

[1] C. Jensen, D. Lin, and B. Ooi, “Continuous clustering of moving objects,” *IEEE Transactions on Knowledge and Data Engineering*, pp. 1161–1174, 2007.

[2] P. Kalnis, N. Mamoulis, and S. Bakiras, “On discovering moving clusters in spatio-temporal data,” *Advances in Spatial and Temporal Databases*, pp. 364–381, 2005.

[3] H. Jeung, M. Yiu, X. Zhou, C. Jensen, and H. Shen, “Discovery of convoys in trajectory databases,” *Proceedings of the VLDB Endowment*, vol. 1, no. 1, pp. 1068–1080, 2008.

[4] J. Gudmundsson and M. van Kreveld, “Computing longest duration flocks in trajectory data,” in *Proceedings of the 14th annual ACM international symposium on Advances in geographic information systems*. ACM, 2006, p. 42.

[5] M. Benkert, J. Gudmundsson, F. Hubner, and T. Wolle, “Reporting flock patterns,” *Computational Geometry*, vol. 41, no. 3, pp. 111–125, 2008.

[6] M. Vieira, P. Bakalov, and V. Tsotras, “On-line discovery of flock patterns in spatio-temporal data,” in *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. ACM, 2009, pp. 286–295.

[7] P. Laube, M. Kreveld, and S. Imfeld, “Finding REMO - detecting relative motion patterns in geospatial lifelines,” *Developments in Spatial Data Handling*, pp. 201–215, 2005.

[8] T. Uno, M. Kiyomi, and H. Arimura, “Lcm ver. 3: Collaboration of array, bitmap and prefix tree for frequent itemset mining,” in *Proceedings of the 1st international workshop on open source data mining: frequent pattern mining implementations*. ACM, 2005, pp. 77–86.

[9] A. O. C. Romero, “Mining moving flock patterns in large spatio-temporal datasets using a frequent pattern mining approach,” Master’s thesis, Master Thesis, University of Twente, 2011.

[10] Q. Li, Y. Zheng, X. Xie, Y. Chen, W. Liu, and W. Ma, “Mining user similarity based on location history,” in *Proceedings of the 16th ACM SIGSPATIAL international conference on Advances in geographic information systems*. ACM, 2008, pp. 1–10.

[11] H. Jeung, H. T. Shen, and X. Zhou, “Convoy queries in spatio-temporal databases,” in *Data Engineering, 2008. ICDE 2008. IEEE 24th International Conference on*, 2008, pp. 1457–1459.

[12] M. Ester, H. Kriegel, J. Sander, and X. Xu, “A density-based algorithm for discovering clusters in large spatial databases with noise,” in *Proc. KDD*, vol. 96, 1996, pp. 226–231.

[13] B. Goethals. (2004) The fimi’04 homepage. [Online]. Available: http://fimi.cs.helsinki.fi/

[14] T. Brinkhoff, “A framework for generating network-based moving objects,” *GeoInformatica*, vol. 6, no. 2, pp. 153–180, 2002.

[15] T. Birkhoff. (2010) Network-based generator of moving objects. [Online]. Available: http://www.fh-oow.de/institute/iapg/personen/brinkhoff/generator/

[16] C. Varschen and P. Wagner, “Microscopic modeling of passenger transport demand based on time-use diaries,” in *Integrated micro-simulation of land use and transport development. Theory, concepts, models and practice*, K. J.

Beckmann, Ed., vol. 81, 2006, pp. 63–69, available at http://elib.dlr.de/45058/. Accessed December 2011.

[17] D. Krajzewicz, G. Hertkorn, C. Rössel, and P. Wagner, "SUMO (Simulation of Urban MObility)," in *Proc. of the 4th Middle East Symposium on Simulation and Modelling*, 2002, pp. 183–187.

[18] MiD2002 Project. (2002) Mobility in Germany 2002. http://daten.clearingstelle-verkehr.de/196/. Accessed October 2011.

[19] SUMO Project. (2011) TAPAS Cologne Scenario. http://sourceforge.net/apps/mediawiki/sumo/index.php?title=Data/Scenarios/TAPASCologne. Accessed October 2011.

[20] Microsoft Research Asia. (2010) Geolife gps trajectories. [Online]. Available: http://research.microsoft.com/en-us/downloads/b16d359d-d164-469e-9fd4-daa38f2b2e13/default.aspx