



Universidad del Bío-Bío
Facultad de ciencias empresariales
Magister en ciencias de la computación

Estrategias de caching semántico geográfico y colaborativo para implementar un LBS en una MANET

Tesis para optar al grado de magíster en ciencias de la computación

Rudyard Guy Fuster Fuster

rudyard.fuster1401@egresados.ubiobio.cl

Profesor Co-Guía

Patricio Galdames

pgaldames@ubiobio.cl

Profesor Guía

Cristian Durán

crduran@ubiobio.cl

21 de Septiembre de 2023

Resumen

Los servicios basados en ubicación o LBS nos ofrecen una serie de servicios para facilitar nuestra vida diaria, entre los que destacan la información del lugar en donde los usuarios se encuentran. Cuando un usuario consulta por un lugar, este entrega su ubicación actual al LBS, comprometiendo su privacidad de ubicación y su privacidad de consulta. El mal uso de la información entregada sin el consentimiento de la persona, puede generar un problema de confianza entre el usuario y el LBS. Existen varios estudios que han intentado proteger la privacidad de ubicación y la privacidad de consulta, los cuales usan distintas técnicas para anonimizar la consulta. Al usar estas técnicas el usuario accede al LBS, esto además de implicar un riesgo de la privacidad, significa una mayor carga de trabajo sobre el LBS. Nuestra propuesta es crear un LBS que opere en una red MANET o red ad hoc inalámbrica conformada por los mismos usuarios. Cuando un usuario realiza una consulta, este pregunta primero a la MANET y en el caso de no encontrar respuesta, se procede a acceder al LBS utilizando técnicas de anonimato. Para el funcionamiento de la red MANET, no se presentan estudios que utilicen una colaboración entre los usuarios sin el uso de repositorios externos de consultas. En la presente tesis, se propone técnicas de caching semántico y colaborativo, para implementar un servicio LBS en una MANET sin el uso de infraestructuras externas. Además se propone como hipótesis que un caché colaborativo, donde cada usuario puede entregar parte de la información a otros usuarios, va a permitir una reducción del acceso al LBS y un mejor uso de memoria de los usuarios móviles. Se proponen dos protocolos de caching colaborativo donde se produce una entrega de datos entre los usuarios de la red y se compararon con sistema de caching individual. Los resultados muestran que ambos protocolos presentan en la mayoría de los escenarios (utilizando variaciones de número de usuarios, velocidades y tamaños de cache) una reducción del acceso al LBS y un mejor uso (menor redundancia) de la memoria en comparación al sistema de caching individual.

Abstract

Location-Based Services (LBS) offer a range of conveniences in our daily lives, prominently providing users with location-specific information. However, this convenience comes at the cost of compromising location and query privacy when users divulge their current whereabouts to LBS providers. Unauthorized sharing of this sensitive information erodes trust between users and LBS providers, underscoring the need for enhanced privacy safeguards. Numerous studies have sought to address the issues of location and query privacy, employing diverse techniques to obfuscate queries. While mitigating privacy concerns, these techniques inadvertently burden LBS providers with heightened computational loads. Our research introduces an innovative LBS paradigm within a Mobile Ad Hoc Network (MANET) – a wireless network formed by interconnected users. When a user initiates a query, the MANET becomes the first point of inquiry. Only when the MANET yields no results does the user use anonymity-preserving methods to access the LBS server. Significantly, our approach distinguishes itself by facilitating MANET cooperation among users without relying on external query repositories. The thesis proposes integrating semantic and collaborative caching techniques to establish an LBS within a MANET, circumventing the need for external infrastructures. Furthermore, we present a novel collaborative caching concept wherein users share query responses, reducing LBS access frequency and optimizing mobile memory utilization. To validate our methodology, we introduce two collaborative caching protocols that facilitate data exchange between network users, comparing their performance against an individual caching system. Empirical results underscore the efficacy of both protocols across varying scenarios involving user counts, speeds, and cache sizes. In most instances, these protocols consistently exhibit reduced LBS access rates and superior memory utilization, underscoring their potential for enhancing the LBS experience within MANETs.

Índice general

1	Introducción	7
2	Conceptos preliminares	9
2.1	Servicios basados en ubicacion (LBS)	9
2.1.1	Encubrimiento de consultas	9
2.1.1.1	k-anonimato	10
2.1.1.2	<i>l</i> -diversidad	10
2.1.1.3	Tabla de Frecuencias de las Consultas	11
2.1.2	MANET	11
2.1.3	Caching colaborativo	13
2.1.4	Caching Semántico	13
2.1.5	Métricas	14
2.1.5.1	Costo de comunicación	14
2.1.5.2	Redundancia	14
2.1.5.3	Tasa de aciertos	14
3	Estado del Arte	15
3.1	Trabajo relacionado	15
4	Hipótesis y Objetivos de la Investigación	18
4.1	Problemática	18
4.2	Hipótesis	18
4.3	Objetivos de la investigación	19
4.3.1	Objetivo general	19
4.3.2	Objetivos específicos	19
4.3.3	Alcance de la investigación	19

5 Metodología	20
5.1 Gestión de Cache	20
5.2 Diseño ciudad	21
5.3 Sistema de consulta	22
5.3.1 Consultas de Nodo	24
5.3.2 Rangos de consulta	25
5.3.2.1 Sistema de consulta por rangos	26
5.3.3 Protocolo de Salida de un Nodo de una Celda	27
5.3.3.1 Protocolo de Salida 1	28
5.3.3.2 Protocolo de Salida 2	29
5.3.3.3 Diferencias entre protocolos	29
5.4 Sistemas propuestos	30
5.4.1 Sistema Individual (SI)	30
5.4.2 Sistema Colaborativo 1 (SC1)	30
5.4.3 Sistema Colaborativo 2 (SC2)	30
6 Resultados y Discusión	31
6.1 Configuración de ciudad simulada	31
6.1.1 Elección de gestión de caché	32
6.2 Resultados experimentales	32
6.2.1 Experimentos con variación de Caché	33
6.2.1.1 Sistemas utilizando consulta por celda	33
6.2.1.2 Sistemas utilizando consulta por rango	35
6.2.2 Experimentos con variación de velocidades	37
6.2.2.1 Sistemas utilizando consulta por celda	37
6.2.2.2 Sistemas utilizando consulta por rango	39
6.2.3 Experimentos con variación de número de nodos	41
6.2.3.1 Sistemas utilizando consulta por celda	41
6.2.3.2 Sistemas utilizando consulta por rango	43
6.2.4 Comparación general de sistemas respecto a resultados experimentales	44
6.2.5 Casos particulares de grandes valores en consultas por rango	45
6.2.5.1 Solución en caso de grandes valores en consultas de rango	46
7 Conclusiones y Trabajo futuro	48
7.1 Conclusiones	48
7.2 Trabajo futuro	51

8 Anexo	55
8.1 Experimentos con amplias variaciones de Caché	55
8.1.1 Sistemas utilizando consulta por celda	56
8.1.2 Sistemas utilizando consulta por rango	57
8.2 Experimentos con amplias variaciones de velocidades	58
8.2.1 Sistemas utilizando consulta por celda	59
8.2.2 Sistemas utilizando consulta por rango	60
8.3 Experimentos con amplia variación de número de nodos	61
8.3.1 Sistemas utilizando consulta por celda	62
8.3.2 Sistemas utilizando consulta por rango	63

Índice de figuras

2.1	Arquitectura de anonimizador.	10
2.2	Arquitectura de una MANET con enmascaramiento de l-diversidad.	12
5.1	División de mapa en grillas con sus puntos de interés.	22
5.2	Rangos consulta generados con diferentes radios	26
5.3	Esquema de comunicación de nodo que sale de celda	28
5.4	Protocolo de Salida 1	29
5.5	Protocolo de Salida 2	29

Índice de cuadros

6.1	Tipos de gestión caché y número de respuestas	32
6.2	Tamaños caché, Sistema individual por celda	33
6.3	Tamaños caché, Sistema colaborativo 1 por celda	34
6.4	Tamaños caché, Sistema colaborativo 2 por celda	34
6.5	Tamaños caché, Sistema individual por rango	35
6.6	Tamaños caché, Sistema colaborativo 1 por rango	36
6.7	Tamaños caché, Sistema colaborativo 2 por rango	36
6.8	Velocidad, Sistema individual por celda	37
6.9	Velocidad, Sistema colaborativo 1 por celda	38
6.10	Velocidad, Sistema colaborativo 2 por celda	38
6.11	Velocidad, Sistema individual por rango	39
6.12	Velocidad, Sistema colaborativo 1 por rango	39
6.13	Velocidad, Sistema colaborativo 2 por rango	40
6.14	Número de Nodos, Sistema individual por celda	41
6.15	Número de nodos, Sistema colaborativo 1 por celda	41
6.16	Número de Nodos, Sistema colaborativo 2 por celda	41
6.17	Número de Nodos, Sistema individual por rango	43
6.18	Número de Nodos, Sistema colaborativo 1 por rango	43
6.19	Número de Nodos, Sistema colaborativo 2 por rango	43
8.1	Anexo: Tamaños caché, Sistema individual por celda	56
8.2	Anexo: Tamaños caché, Sistema colaborativo 1 por celda	56
8.3	Anexo: Tamaños caché, Sistema colaborativo 2 por celda	56
8.4	Anexo: Tamaños caché, Sistema individual por rango	57
8.5	Anexo: Tamaños caché, Sistema colaborativo 1 por rango	57
8.6	Anexo: Tamaños caché, Sistema colaborativo 2 por rango	57
8.7	Anexo: Velocidad, Sistema individual por celda	59

8.8 Anexo: Velocidad, Sistema colaborativo 1 por celda	59
8.9 Anexo: Velocidad, Sistema colaborativo 2 por celda	59
8.10 Anexo: Velocidad, Sistema individual por rango	60
8.11 Anexo: Velocidad, Sistema colaborativo 1 por rango	60
8.12 Anexo: Velocidad, Sistema colaborativo 2 por rango	60
8.13 Anexo: Número de Nodos, Sistema individual por celda	62
8.14 Anexo: Número de nodos, Sistema colaborativo 1 por celda	62
8.15 Anexo: Número de Nodos, Sistema colaborativo 2 por celda	62
8.16 Anexo: Número de Nodos, Sistema individual por rango	63
8.17 Anexo: Número de Nodos, Sistema colaborativo 1 por rango	63
8.18 Anexo: Número de Nodos, Sistema colaborativo 2 por rango	63

Capítulo 1

Introducción

Gracias al desarrollo tecnológico de los estos últimos años, el uso de aplicaciones móviles es parte del día a día de todas las personas para poder realizar su labores de trabajo y recreativas con normalidad. En este ámbito, los Servicios Basados en Ubicación (conocido por siglas en inglés Location-Based Services o LBS) (1), nos ofrecen información de los puntos de interés (por su sigla en inglés Point of Interest o POI) que queremos visitar en base a la ubicación geográfica del usuario, como lo hacen las aplicaciones. Ejemplo de estos servicios es Google maps, el cual requiere la posición geográfica del usuario para generar un mapa de los puntos de interés alrededor de este usuario. Además, estos servicios nos pueden indicar otros POIs de la misma temática que estén cercanos. Por ejemplo, un usuario de una aplicación desea buscar restaurantes que estén a un rango de distancia de dos cuadras de su paradero, pero el LBS no encuentra este tipo de POI dentro de estos rangos. Sin embargo, si existen bares que se le podrían ser recomendados por la similaridad semántica, ya que un bar se puede asociar a un sitio en donde se consume alimentos, es decir es un sitio de características similares. Esto último corresponde al uso de la semántica en los servicios basados en ubicación.

Hay que tener en cuenta que el uso de un LBS puede traer problemas al manejar la ubicación del usuario. El principal es la problemática de la confianza que el usuario puede entregarle al LBS, debido a la posibilidad de un mal uso de la información entregada sin consentimiento. Este mal uso puede ser el de empresas que no informan al usuario para qué utilizan la información, la cual puede ser vendida a terceros, o también puede ocurrir un ataque al LBS y que esta información pueda ser filtrada. El servidor LBS puede almacenar información del usuario de los lugares que ha visitado o los que ha consultado, si un tercero obtiene esta información puede generar un peligro para el usuario. Por ejemplo, si el usuario consultó ubicaciones de clínicas o farmacias cercanas, se puede inferir que la persona pade-

ce de alguna enfermedad y se podría utilizar esta información para el beneficio de alguna empresa, o en el caso que el usuario consulte por lugares de vacaciones en una fecha en específico, se puede inferir que dejará su casa inhabitada.

Existen varios enfoques para proteger la privacidad de ubicación y de consulta del usuario, en su mayoría se basan en la utilización de anonimizadores para enmascarar la consulta al LBS usando técnicas de k -anonimato para proteger la privacidad de ubicación (2) y l -diversidad (3) para proteger la privacidad de consulta. Si bien estas técnicas reducen el riesgo de pérdida de privacidad, no lo eliminan del todo. Mientras se contacte con el LBS una menor cantidad de veces, será más beneficioso para el usuario, ya que se protege su privacidad y a la vez también puede serlo para el LBS al reducir la carga de trabajo que se genera al responder consultas simultáneas. Para evitar el contacto con el LBS, el cual se convierte en nuestro “adversario”, en este trabajo de tesis, proponemos crear un servicio LBS que opere sobre una MANET. Podemos explicar el concepto de una MANET de la siguiente manera: Si una persona se encuentra en un lugar desconocido de la ciudad, pregunta por una dirección a las personas de alrededor para obtener una respuesta o el camino seguir para encontrar el lugar que está buscando. Si se sigue este mismo esquema pero siendo en lugar de personas reales, dispositivos móviles que preguntan a otros dispositivos en un cierto rango de distancia, podemos estar hablando de una MANET. Aquí los dispositivos hacen consultas entre ellos sin contactar con una entidad externa como lo es el servidor LBS. La contribución de la presente propuesta se basa en el diseño de técnicas de caching semántico colaborativo entre los usuarios de una MANET, protegiéndose del LBS en términos de privacidad de ubicación y privacidad de consulta. Se debe tomar en cuenta que en esta propuesta, se confía en los usuarios de la MANET.

El presente informe de tesis está organizado de la siguiente manera; En el Capítulo 2 se habla de conceptos preliminares a tener en cuenta para el desarrollo de esta investigación. En el Capítulo 3 se expone el estado del arte en que se encuentra esta temática. En el Capítulo 4 se presenta la problemática, la hipótesis, los objetivos y el alcance de la investigación. En el Capítulo 5 se explica la metodología a usar en la investigación. En el capítulo 6 se presentan los resultados experimentales. En el Capítulo 7 se muestran las conclusiones y trabajos futuros. Finalmente en el Capítulo 8 se presentan los Anexos de la investigación donde se incluyen más resultados experimentales.

Capítulo 2

Conceptos preliminares

2.1 Servicios basados en ubicación (LBS)

Como su nombre lo indica, corresponden a servicios basados en la ubicación geográfica de la persona. El usuario realiza una consulta basada en su ubicación a un servidor LBS, se le llama LBQ (i.e., por su sigla en inglés Location-Based Queries) (1). Esta LBQ es cualquier consulta geográfica relativa a la posición del usuario, que contiene los datos de ubicación y de la consulta que se está realizando. Esta consulta también puede estar acotada por rango de distancia a la ubicación del usuario, como lo es por ejemplo la búsqueda de hoteles en un rango de tres cuadras de mi ubicación geográfica.

2.1.1 Encubrimiento de consultas

Cuando el usuario envía una consulta al LBS, lo usual es que se utilice un anonimizador que proteja su consulta (4), a esto también se le puede llamar “encubrimiento o enmascaramiento de la consulta”. En Figura 1, se puede apreciar cómo funciona: primero el usuario envía una LBQ al anonimizador (1), luego de pasar por este, la consulta queda enmascarada (se le aplica k -anonimato y l -diversidad) y es enviada al servidor LBS (2), el servidor entrega las “respuestas candidatas” (este concepto será explicado en la siguiente sección) (3), para luego que el anonimizador filtre las respuestas y entregue la consulta resultante al usuario (4). El anonimizador puede trabajar desde dos perspectivas, proteger la privacidad de ubicación, siendo la técnica más usada el del k -anonimato, y desde la perspectiva de proteger la privacidad de la consulta, siendo la más usada la técnica de l -diversidad.

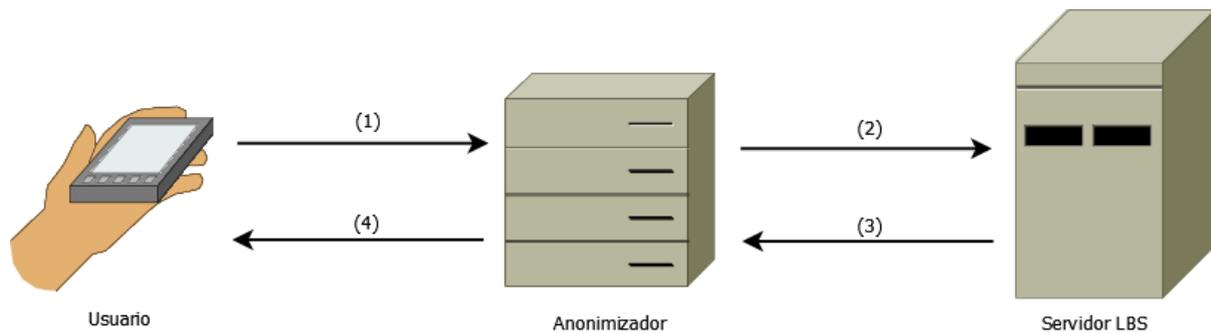


Figura 2.1: Arquitectura de anonimizador.

2.1.1.1 k-anonimato

Esta técnica (5) se basa en escoger $k-1$ ubicaciones posibles para enviarle un LBQ al servidor LBS. Estas ubicaciones son escogidas en base a la probabilidad que existan usuarios que se mueven por esta zona. Luego son enviadas consultas simultáneas al LBS con $k-1$ ubicaciones distintas, con lo que se impide que el LBS conozca la ubicación real del usuario. A estas $k-1$ ubicaciones también se les llama ubicaciones factibles (6). Las respuestas que entrega el LBS de todas estas consultas, se les llaman “respuestas candidatas”, ya que se incluye la respuesta de la consulta real, más las respuestas de las otras consultas factibles, ya que es posible que la solicitud de LBS de un usuario pueda ser combinada con otras realizadas en el mismo momento por otros usuarios. Por ejemplo, un usuario desea saber dónde hay tiendas de volantines en su zona, cuando envía la consulta, la cual contiene su ubicación, la recibe un anonimizador que utiliza k -anonimato, supongamos que se escoge un $k=4$, es decir 4 ubicaciones, luego de escoger estas, se envían 5 consultas de tiendas al LBS en base a 5 ubicaciones distintas más la del usuario, el LBS responde estas 5 consultas sin saber cual pertenece al usuario, luego del posterior filtrado efectuado por el anonimizador la persona obtiene la respuesta a su consulta.

2.1.1.2 l -diversidad

El ejemplo anteriormente dado sobre el k -anonimato, protege al usuario de su ubicación, pero se mantiene el conocimiento de la consulta. Con esta información se podría, y siguiendo con el ejemplo del párrafo anterior, descartar algunas k ubicaciones, quizás una de estas 4 ubicaciones del ejemplo sea en una estación de metro o en un estadio, lugares que es difícil pensar en que una persona compre un volantín, pero si entre estas ubicaciones se encuentra un parque o una playa, estas podrían ser las ubicaciones reales y se descartarían las otras, aumentando más la probabilidad de detección de ubicación de la persona. La

l -diversidad (7) puede ayudar a dar solución a esta problemática, esta consiste en escoger l consultas semánticamente diferentes a la consulta del usuario, estas se escogen en base a que tengan una probabilidad de consulta similar a la del usuario en esa zona. Todas estas consultas son enviadas al LBS de manera simultánea. Para la ejecución de la l -diversidad, las consultas enviadas se combinan con k -anonimato (2). Siguiendo con el mismo ejemplo, en el caso de que se desea saber sobre tiendas de volantines que están en mi zona, aplicamos l -diversidad y supongamos que se escoge un $k=4$ y un $l=3$, esto quiere decir que se van a escoger 4 ubicaciones y en cada una de estas se elige entre tres tipos de consulta con términos semánticamente distintos, como lo puede ser farmacias, restaurantes y bancos. Finalmente se están enviando 5 consultas, contando la del usuario, desde 5 ubicaciones distintas y con un contenido distinto. Siguiendo con el ejemplo anterior, en el caso de ubicaciones como estaciones de metro o estadios, ya no van a poder ser fácilmente descartadas, porque ahora no solo se están buscando tiendas de volantines, también pueden existir consultas de otras temáticas en esta ubicación. Al igual que en el k -anonimato, luego de un posterior filtrado de las respuestas del LBS, el usuario obtiene la respuesta de su consulta, esta vez con privacidad de ubicación y el contenido de la consulta del usuario.

2.1.1.3 Tabla de Frecuencias de las Consultas

La Tabla de Frecuencias de las Consultas, también conocida como QFT (i.e., por su sigla en inglés Query Frequency Table), corresponde a una estructura de datos (2) en donde se almacenan el número de consultas enviadas desde una ubicación en específico. Esta es usada para la ejecución de técnicas de anonimato. Por ejemplo, al aplicar la l -diversidad, las consultas “factibles” escogidas, son seleccionadas en base a lo registrado en la QFT de cada división geográfica o celda (3) (el concepto de “celda” será explicado mas adelante).

2.1.2 MANET

El uso de las técnicas anteriormente mencionadas, si se utilizan en una multitud de consultas simultáneas de distintos usuario suele causar un cuello de botella en el LBS, haciendo que todo el proceso se ralentice. Una manera de solucionar este problema es usar una MANET o red móvil ad-hoc inalámbrica, la cual como su nombre lo indica es un tipo de red peer to peer (P2P) inalámbrica distribuida que está creada por distintos dispositivos móviles. A cada uno de estos dispositivos móviles se les llama nodo. Al utilizar una MANET, además de evitar este cuello de botella, se busca principalmente evitar el contacto del usuario con el LBS, ya que cuando este usuario o nodo desea hacer una consulta, primero la efectúa a sus nodos

aledaños, es decir, se intenta resolver la consulta dentro de la MANET creada, para luego, en caso de no encontrar respuesta (si estos nodos no poseen la respuesta), se le envía la consulta al LBS con los correspondientes enmascaramientos de consulta. Esto se puede apreciar en la Figura 2, en la cual primero el usuario envía una consulta a los vecinos de la red MANET (1), esta consulta es repetida entre los nodos de la red (2) y (3) para saber si alguien tiene la respuesta. En el caso de no encontrarse respuesta en la entre los nodos, se envía una consulta enmascarada al LBS (4), en el caso de la figura es utilizando l-diversidad, luego se entregan las l-respuestas candidatas (5), por motivos prácticos se omite en la Figura 2 el posterior filtrado del anonimizador explicado en la Figura 1, para que finalmente se entregue la respuesta de la consulta al usuario (6).

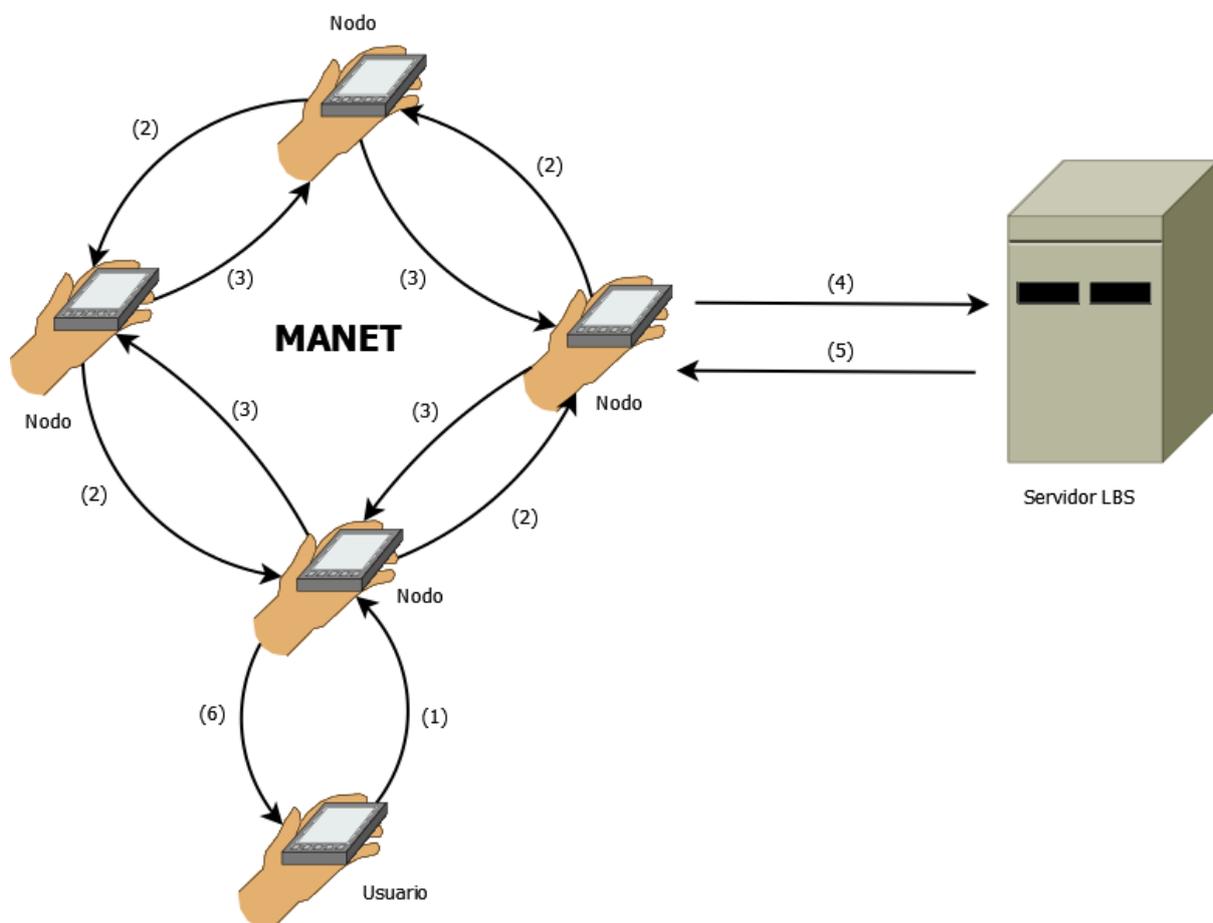


Figura 2.2: Arquitectura de una MANET con enmascaramiento de l-diversidad.

A diferencia de otros servicios de redes de comunicación, como es el caso de servicios de telefonías, la MANET no posee infraestructuras de apoyo como lo podrían ser antenas

de telefonía, por lo que la información no es almacenada en repositorios externos, solo los usuarios almacenan la información de la consulta. Esto si bien puede limitar el alcance de la arquitectura, y el tiempo de funcionamiento debido a la batería de los dispositivos, entrega una mayor privacidad al usuario, ya que su consulta no pasa el LBS si los usuarios en las cercanías poseen la información que se busca.

2.1.3 Caching colaborativo

La memoria caché corresponde a la memoria interna que posee un dispositivo, como lo puede ser un teléfono, una tablet, o un notebook con Wifi. En esta memoria se guardan los datos que pueden usarse en futuras consultas, el término de caching viene de “almacenamiento en memoria caché”. Para gestionar la memoria distribuida presente en los dispositivos, se tiene que tomar en cuenta lo siguiente:

- La información geográfica este cercana al lugar que le corresponda.
- Se intente reducir la redundancia.
- Se limite el costo de comunicación.

Las técnicas de caching colaborativo (6), buscan balancear estas tres métricas.

2.1.4 Caching Semántico

El concepto de semántica corresponde a un determinado conjunto que cumple ciertas propiedades en particular (8) y (9). En nuestro modelo de caching semántico, este conjunto va a corresponder a un tipo de punto de interés en específico que el usuario quiere conocer (hoteles, hospitales, restaurantes, etc.). Es decir se hará búsqueda del tipo exacto de punto de interés. Se dejará como trabajo futuro el uso del concepto de “similaridad semántica”, el cual corresponde a entregar puntos de interés de una temática similar como respuesta válida. Un ejemplo de esto último, es cuando un usuario pregunte por hospitales en el área, se le responda que si bien no hay hospitales, existen farmacias en la zona, por lo cual se podría entregar esta información, como una respuesta equivalente. Existen estudios, como el de (10), donde se establecen listados de términos más comunes que buscan los turistas, los cuales pueden ayudar a establecer conexiones de similaridad semántica.

2.1.5 Métricas

En la literatura existen muchas métricas para analizar el funcionamiento de técnicas que usan caché semántico colaborativo, en esta propuesta se piensan usar en específico tres:

2.1.5.1 Costo de comunicación

Se refiere al número de mensajes intercambiados en la MANET. Cuando se usa esta métrica además de medirla con respecto al tiempo también mide en relación al parámetro k consultas usado para el k -anonimato y l -diversidad (11).

2.1.5.2 Redundancia

Corresponde a la cantidad de puntos de interés repetidos en la caché de usuario. Esto puede ser vista a nivel de la ciudad y a nivel de celda. En esta última, la redundancia es medida con respecto a los puntos de interés en las caches de usuarios pertenecientes a una celda en específica.

2.1.5.3 Tasa de aciertos

Corresponde a la tasa de aciertos del caché (6). En otras palabras, cuantas veces se encuentra la respuesta en la memoria caché del usuario y de los vecinos de la MANET cuando el usuario envía una consulta. Esto se puede apreciar en la Ecuación 2.1 .

$$\text{Tasa de aciertos del caché} = \frac{\text{Número de aciertos del caché}}{\text{Número de aciertos del caché} + \text{Número de desaciertos del caché}} \quad (2.1)$$

Capítulo 3

Estado del Arte

3.1 Trabajo relacionado

Dentro del área del caching colaborativo se distinguen estudios principalmente en dos áreas de investigación, caching colaborativo en servicios LBS esporádicos (o simplemente servicios LBS) y LBS continuos, estos últimos corresponden a sistemas de caching colaborativo en donde los usuarios móviles tienen una trayectoria definida, es decir un origen y una dirección final predeterminada. En los LBS continuos, además las respuestas de las consultas van a ir cambiando a medida que se va generando el movimiento (existen puntos de interés móviles).

En el área de investigación de LBS continuos se destaca el estudio de (12) donde se propone una arquitectura de caché colaborativo a través de una red P2P para mitigar la degradación de rendimiento, creando un filtro que descarta las respuestas candidatas y actualizando solo el resultado real. También se destaca el estudio de (13), donde se utilizan modelos de Markov para predecir la siguiente consulta de acuerdo con la movilidad del usuario, y así crear un algoritmo para efectuar el k-anonimato en el caso de no encontrar respuesta en el caché. En el estudio de (14), se crea un sistema de caching colaborativo que solo necesita la siguiente ubicación del usuario para predecir toda la trayectoria del usuario. Cabe destacar el estudio de (15) sobre LBS continuos, el cual si bien no es de caching colaborativo trabaja sobre un sistema de caché de dos niveles, donde el primer nivel corresponde al caché del usuario y el segundo nivel corresponde al caché del anonimizador, el cual también almacena las respuestas de las consultas.

En el área investigación de LBS esporádicos, cuya área pertenece a esta propuesta, destaca principalmente el estudio de (6) donde se crean repositorios, llamados puntos de acceso

en cada celda de la ciudad simulada, en estos repositorios se almacenan las respuestas las consultas de los usuarios, incluyendo las respuestas candidatas. Esto funciona como una especie de caché externo en donde los usuarios pueden consultar por POIs. En el estudio de (16), se propone un esquema de caching de consultas espaciales con el apoyo de drones, que funcionan también a modo de repositorios externos, estos drones ayudan además en la generación de k ubicaciones para el uso del k -anonimato. En el estudio de (17) se presenta una idea de caching colaborativo de 3 caches, que corresponden al caché del usuario, y dos puntos de acceso, divididos en caché de las respuestas de usuarios en la misma celda y de consultas pendientes de respuesta en la misma celda. En el estudio de (1) se propone un esquema de caché colaborativo en donde en el caso de no encontrar respuesta en el caché de los vecinos, se escoge un líder en basado en el número de conexiones que ha efectuado con el LBS y el número de respuestas en su caché para que sea este el que envíe la consulta enmascarada al LBS. Finalmente en el estudio de (18) se ofrece un sistema que combina caching colaborativo y k -anonimato donde se consideran las diferencias entre las frecuencias de consultas en diferentes horas del día.

Cabe mencionar que dentro del uso de las MANET, existe también el concepto de VANET o redes vehiculares ad-hoc inalámbricas el cual es un tipo de MANET, pero en vez de dispositivo móviles usamos vehículos con el mismo fin, como nos muestran los trabajos de (19) y (20). En este ámbito estos nodos poseen distintos tipos de problemáticas como el de la variabilidad de conexión o el de las diferentes velocidades de los vehículos en el mapa con sus diferentes trayectorias de posición.

En el área de la mejora de rendimiento de la mensajería y las consultas generadas en una MANET cabe destacar el estudio de (21) que busca crear un esquema de caching para una CCMANET (Content Concentric MANET) para reducir el valor del tramo entre los nodos de la MANET y la carga del servidor. En (22) se propone un modelo de caché colaborativo que busca reducir la distancia de transmisión de los mensajes en una MANET. Finalmente en (23) se propone un esquema de caching colaborativo que busca clasificar las solicitudes de consulta en la MANET para reducir la sobrecarga de la caché.

Como se puede apreciar ningún estudio ha trabajado en el almacenamiento de las respuestas candidatas entregadas por el LBS en el caché de los usuarios, siendo el estudio más cercano a esta propuesta el de (6), pero el cual almacena estas respuestas candidatas en un punto de acceso externo al usuario en cada celda, a diferencia de nuestra propuesta donde

se utiliza solo la caché local. A la hora de recibir las respectivas respuestas del LBS, múltiples usuarios podrían almacenar en su caché las respuestas candidatas entregadas ya que si bien no corresponden a una información que el usuario está buscando, son datos válidos y correctos que algún otro usuario podría requerir en un futuro.

Capítulo 4

Hipótesis y Objetivos de la Investigación

4.1 Problemática

Los modelos propuestos anteriormente de redes MANET, son de carácter individual, es decir cada nodo efectúa una consulta al LBS, luego de efectuar las respectivas técnicas de anonimato, para recibir una respuesta que almacenará en su caché. Además, no se presentan estudios que utilicen una colaboración entre los nodos sin el uso de repositorios externos de consultas, es decir, no hay estudios que utilicen una colaboración usando exclusivamente la información en la caché de los nodos para responder consultas.

En el área de tipos de gestión de caché, no ha sido explorada en la literatura en profundidad, principalmente se utiliza la eliminación por antigüedad (6) y por distancia (8) y (24). Creemos que pueden existir otros sistemas de reemplazo de caché que pueden optimizar el uso de esta, y también se puede demostrar bajo ciertos parámetros que sistema de gestión de caché pueden ser más óptimos.

4.2 Hipótesis

Un caché colaborativo va a permitir una reducción del acceso al LBS y por lo tanto, una mayor protección de la privacidad de ubicación en comparación con el uso de un caché local, donde cada nodo decide como gestionar su propio caché. Se espera que las técnicas a desarrollar tomen en cuenta:

- Nodos se coordinan en qué almacenar, sin generar un alto costo de comunicación.
- Múltiples nodos aprovechen la respuesta de una consulta.

4.3 Objetivos de la investigación

En esta sección se presenta los objetivos generales y específicos de la investigación.

4.3.1 Objetivo general

Evaluar el rendimiento de técnicas de caching semántico y colaborativo, para implementar un servicio LBS en una MANET que sea consiente en términos de privacidad de ubicación y de consulta.

4.3.2 Objetivos específicos

Los objetivos específicos de la investigación son:

1. Selección técnicas de caching semántico y colaborativo entre usuarios de una MANET.
2. Incorporación de diversas técnicas de caché semántico colaborativo en una MANET.
3. Evaluar el rendimiento de las técnicas empleadas con respecto a las otras existentes en la literatura.

4.3.3 Alcance de la investigación

- Cada usuario posee un dispositivo móvil el cual pertenece a la MANET.
- Cada usuario en una celda corresponde a un nodo, todos los nodos son homogéneos dentro del ambiente simulado.
- EL modelo de movimiento de los usuarios, esta creado en base al modelo de random walk. El cual consiste en crear un camino en base a una sucesión de pasos aleatorios (25). El usuario, al estar en movimiento con respecto al tiempo, va a ir variando su posición, por lo que las respuestas de las consultas efectuadas en un primer momento pueden no ser las óptimas al momento de ser recibida la respuesta.
- Existe una cantidad mínima de nodos que permite alcanzar a cualquier nodo en cualquier momento de la red MANET.
- Durante el periodo de simulación, se asume que los datos no sufren obsolescencia, ya que no se estudiarán largos periodos de tiempo.

Capítulo 5

Metodología

5.1 Gestión de Cache

Como la cantidad de memoria caché de cada usuario es limitada, existen diferentes técnicas para mejorar este uso de memoria de la forma más eficientemente posible.

Para analizar cuál es la técnica de gestión de caché más óptima se proponen una serie de estrategias de caching a experimentar para determinar cuál será la más eficiente para usar en la simulación del sistema de caching semántico colaborativo. A continuación se presenta las técnicas a comparar en la simulación:

1. **No hay gestión**

En este caso se medirán resultados para el caso que no se efectúa una gestión del caché, es decir una vez que la memoria caché del usuario se llene, se mantendrá de esta forma hasta al final de la simulación.

2. **Eliminar más antiguo**

Corresponde a utilizar un esquema FIFO en la memoria caché, es decir cuando la memoria se llena, se elimina el primero en agregarse.

3. **Eliminar al azar**

Cuando la memoria caché se llena se eliminará un valor al azar de la caché.

4. **Eliminar el más distante**

Según la distancia del usuario con respecto al POI almacenado en el caché, se va eliminado el dato más alejado a la posición actual del usuario.

5. Eliminar el menos popular

La popularidad corresponde a la frecuencia de consulta, esta se basa sobre qué tan frecuente son las consultas sobre un POI almacenado en el caché del usuario, por parte de los vecinos y del mismo usuario cuando encuentra la respuesta es su caché. Se escoge eliminar el POI que posea un menor número de consultas cuando sea necesario eliminar datos.

6. Resetear la caché

Una vez que la caché de llena, se procede a eliminar todos los datos.

7. Eliminar el más popular

Sistema similar al del punto 5, pero esta vez se escoge eliminar el POI que posea un mayor número de consultas cuando sea necesario eliminar datos.

8. Eliminar menos Popular/Distante

Se crea una relación entre la popularidad y distancia que funciona como se muestra en la ecuación:

$$\text{Razon} = \frac{\text{Número de popularidad del POI}}{\text{Distancia entre el POI y la posición del usuario}} \quad (5.1)$$

Se procede a eliminar el menor valor encontrado.

9. Eliminar más Popular/Distante

Sistema similar al del punto 8, pero esta vez se procede a eliminar el valor de la ecuación resultante, cuando sea necesario eliminar datos.

10. Reseteo caché al entrar en celda nueva

Cada vez que el usuario cambia de celda, procede a eliminar todos sus datos almacenados.

5.2 Diseño ciudad

En la ciudad simulada, el espacio geográfico es discretizado en unidades geográficas básicas o pasos. El espacio es dividido en zonas cuadradas denominadas celdas. El tamaño de cada arista de una celda es de 10 pasos. Dentro de estas celdas se encuentran los puntos de interés que el usuario visita, y además es donde se produce el movimiento de los usuarios los cuales

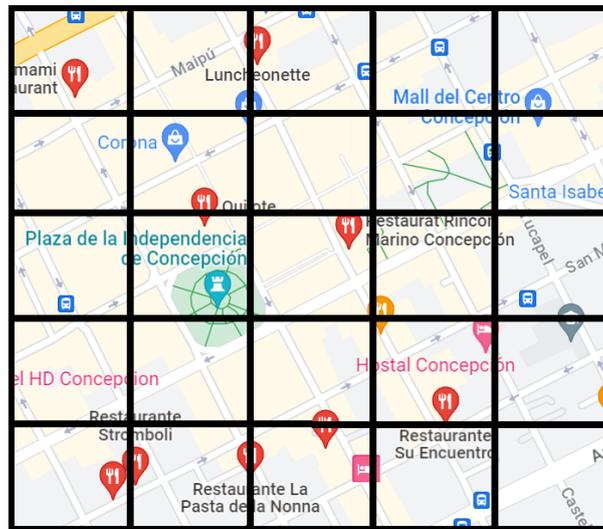


Figura 5.1: División de mapa en grillas con sus puntos de interés.

se pueden mover fuera de estas hacia otras celdas, siguiendo el esquema de movimiento de random walk. En Figura 5.1, se puede apreciar un mapa del centro de la ciudad de Concepción, Chile la cual contiene puntos de interés como restaurantes y tiendas. A este mapa se le aplica una división por celdas, para formar una ciudad simulada de ejemplo.

5.3 Sistema de consulta

A continuación se presentan el funcionamiento de las funciones primitivas usadas para el funcionamiento del sistema:

- **PreguntarMiCache()** = Se consulta en la memoria caché del nodo si tiene almacenado un POI de la misma temática y en la misma celda de la consulta.
- **PreguntarNodos()** = Se consultan a los nodos vecinos si tienen almacenando un POI de la misma temática y en la misma celda de la consulta.
- **PreguntarLBS()** = Se consulta al LBS sobre un POI de la misma temática de la consulta y ubicada en la misma celda del nodo. El sistema está diseñado para que el LBS siempre obtenga la respuesta correcta.
- **ConsultarLider()** = El nodo consulta si existe un líder dentro en la celda, si se retorna false, el nodo se convierte en el líder, y solicita la QFT al antiguo líder de la celda. En este caso, si bien el antiguo líder no posee la condición de líder de esa celda, sigue

en posesión de la QFT la cual es única para cada celda, por lo que funciona como un identificador para que este pueda ser encontrado en cualquier lugar de la ciudad en caso de ser requerido.

- **NuevoLider()** = Si el nodo que cambia de celda es el antiguo líder de la celda antigua, se busca un nuevo candidato en la celda, si se encuentra candidato, este pasa a ser el nuevo líder de la celda y se le entrega la QFT. En caso de no encontrarse candidato, es decir no se encuentran nodos de la celda, la celda pasa a un estado null y el nodo que cambia de celda se llega la QFT de esa celda.
- **EntregarDatos()** = Corresponde a la entrega de datos de caché que puede efectuar un nodo al salir de la celda, esto se realiza llamado a un protocolo de entrega de datos de la caché que será explicado más adelante.

En la ciudad simulada, todos los usuarios efectúan las acciones de: consultar por un tipo puntos de interés, moverse y efectuar un cambio de celda en caso que su movimiento así lo indique. Todos los usuarios son homogéneos entre sí, es decir, tienen el mismo tamaño de memoria caché, y comienzan la simulación con una caché vacía en un punto aleatorio del mapa. Si bien todos los nodos son homogéneos, existen nodos líderes, los cuales tienen como única función extra la de actualizar la QFT, que es usada para efectuar las técnicas de anonimato, como fue precisado en el Capítulo de Conceptos Preliminares.

El sistema de movimiento random walk, consiste en efectuar al azar uno de los cuatro posibles movimientos; arriba, abajo, izquierda, derecha. Este movimiento se produce una vez por segundo mientras se esté ejecutando la simulación. El nodo al entrar en una celda nueva, produce el cambio de celda anteriormente mencionado, en el cual el nodo, en el caso de ser líder de la celda anterior llama a **NuevoLider()**, para luego llamar a **ConsultarLider()** en la nueva celda, como lo muestra el Algoritmo 1. En el caso de no ser el líder de la antigua celda, pasa directamente a **ConsultarLider()**. En el caso que se esté utilizando un sistema colaborativo, al salir de la celda, se procede a realizar los protocolos de entrega de datos que serán explicados más adelante. La salida entregada por el Algoritmo 1 correspondiente a la actualización del estado del Nodo, corresponde a determinar su estado de “líder” o de “no líder”.

Algorithm 1: Cambio de Celda

Data: Posicion de Nodo, EstadoLider**Result:** Actualizacion de estado de Nodo

```
1 if EstadoLider == true then
2   | EstadoLider = false.
3   | Llamar NuevoLider().
4   | Llamar ConsultarLider().
5 else
6   | Llamar ConsultarLider().
7 if Colaboracion == true then
8   | Llamar EntregaDatos().
9 else
10  | END.
```

5.3.1 Consultas de Nodo

El sistema de consultas geográficas es similar al usado en (26), pero en nuestro sistema el usuario solo consulta por un tipo punto de interés perteneciente a la celda a la que se encuentra, y una respuesta válida es exclusivamente el punto de interés del mismo tipo que el de la consulta y pertenece a la celda en específico. Un ejemplo de esto es que una persona pregunte por hoteles en la cuadra en donde se encuentra. En un sistema clásico individual, el usuario para responder una consulta revisa primero si tiene la respuesta en su memoria caché, en caso de no encontrar respuesta, procede a consultar al LBS luego de usar las correspondientes técnicas de anonimato. En nuestro sistema individual de consultas se actúa como se aprecia en el algoritmo 2.

Algorithm 2: Consulta de Nodo

Data: Tipo de POI en una celda
Result: POI de la celda

```

1 if PreguntarMiCache() == false then
2   | if PreguntarNodos() == true then
3   |   | Se almacena el POI en la caché de Usuario.
4   | else
5   |   | PreguntarLBS()
6   |   | Se almacena el POI en la caché de Usuario.
7 else
8   | Consulta respondida.
```

Una vez obtenida la respuesta, es decir el POI, el usuario pasa a almacenar este en su memoria caché. Para un sistema colaborativo, se procede a actuar de manera similar pero bajo la base de que solo se almacenaran datos nuevos para los usuarios de la celda en la caché. Un “dato nuevo”, corresponde a un POI que no está almacenado en ninguna caché perteneciente a los vecinos de la celda, es decir si un vecino tiene la respuesta a la pregunta que está efectuando el usuario, esta respuesta no se almacenará.

5.3.2 Rangos de consulta

El sistema de consultas presentado anteriormente solo pregunta a los vecinos dentro de la celda. La consulta se puede expandir si variamos el rango de esta consulta a más celdas, es decir, se pregunta por un punto de interés a los vecinos en celdas aledañas. Como se muestra en la Figura 5.2, primero se pregunta a los nodos dentro de la celda, si no se encuentra la respuesta, antes de ir al LBS, se consulta en un rango de una circunferencia, que vendrían siendo los vecinos de la celda, y en el caso de no encontrar respuesta, se va a ir aumentando el radio de consulta, es decir ahora se pregunta a los vecinos de los vecinos, y así sucesivamente. El valor del radio es igual $R = \text{Diagonal de la celda}$, para que siempre se incluya toda la celda en la consulta, independiente de la posición en donde se encuentre el nodo. En caso de no encontrar una respuesta correcta a la consulta, se duplica el Radio de consulta como se aprecia en la Figura 5.2. Este sistema se podría seguir ampliándose hasta consultar a toda la ciudad, pero se estima que el costo de comunicación sería enorme, por lo cual se toma la decisión de trabajar con no más de 3 “saltos” o aumentos del Radio de consulta.

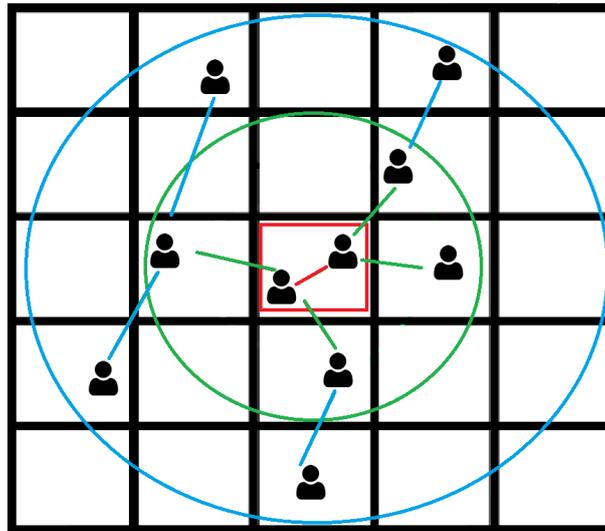


Figura 5.2: Rangos consulta generados con diferentes radios

5.3.2.1 Sistema de consulta por rangos

A continuación se presenta el pseudocódigo del funcionamiento de una consulta por rangos, correspondiente al Algoritmo 3. Se trabaja de la siguiente manera: Primero el nodo consulta en su caché si tiene un POI de la misma temática y en la misma celda en la que se encuentra. En el caso de no encontrarse, pregunta a los vecinos de la celda. En este caso se agrega un parámetro $R=0$, es decir un radio de cero, lo que significa que no se genera una consulta en un rango, solo es una consulta dentro de la celda. En el caso no encontrarse respuesta, si se está trabajando usando rangos (Rangos = true) se pasa a consultas por rangos fuera de la celda, y se va a ir aumentando el Radio de la consulta si se sigue sin encontrar respuesta. Si se encuentra una respuesta afirmativa, automáticamente se termina el ciclo, el cual también está limitado a un Radio máximo. En el caso de seguir sin encontrar respuesta una vez que se alcanza el Radio máximo, se procede a consultar al LBS. En el caso de no trabajar utilizando Rangos, luego de no encontrar respuesta entre los vecinos de la celda, se procede a consultar directamente al LBS. En los casos de consultas las LBS, se asume que se utilizaran las respectivas técnicas de anonimato, antes de generar la consulta.

Algorithm 3: Consulta de Nodo por rangos

Data: Tipo de POI en una celda, Radio, RadioMax
Result: POI de la celda

```

1 ValorEncontrado == false
2 if PreguntarMiCache() == false then
3   if PreguntarNodos(0) == true then
4     Se almacena el POI en la caché de Usuario.
5   else
6     if Rangos == true then
7       while Radio < RadioMax do
8         if PreguntarNodos(Radio) == true then
9           Se almacena el POI en la caché de Usuario.
10          ValorEncontrado = true
11          break.
12         else
13           Radio += Radio.
14       if ValorEncontrado == false then
15         PreguntarLBS().
16         Se almacena el POI en la caché de Usuario.
17     else
18       PreguntarLBS().
19       Se almacena el POI en la caché de Usuario.
20 else
21   Consulta respondida.

```

5.3.3 Protocolo de Salida de un Nodo de una Celda

El sistema colaborativo, utiliza un sistema un protocolo de entrega de datos, esto consiste en que cuando un nodo sale de su celda, se comunica con los nodos de la celda de la cual proviene y entrega la información de los POI que tiene en su caché, que corresponden a los POI ubicados dentro la celda. El nodo que sale se comunica con sus antiguos vecinos, como se aprecia en la Figura 5.3. Cabe mencionar que en un sistema individual de consultas, no se realiza una entrega de datos.

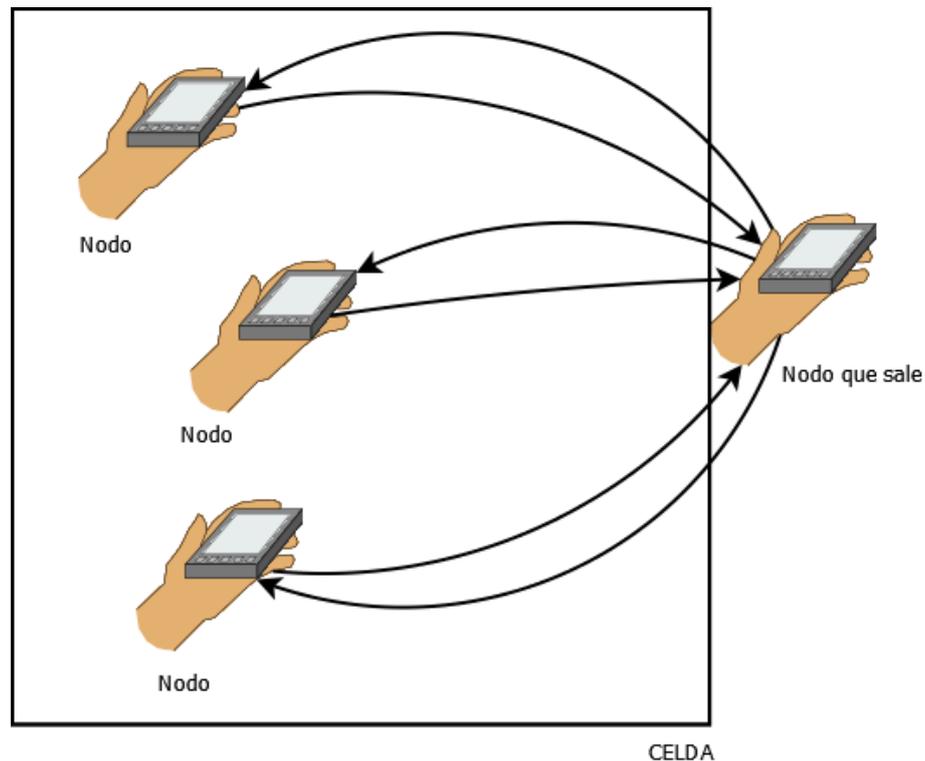


Figura 5.3: Esquema de comunicación de nodo que sale de celda

5.3.3.1 Protocolo de Salida 1

Para la de entrega de datos se crearon dos protocolos. El protocolo número 1, funciona como se aprecia en la Figura 5.4. En esta figura se presenta un esquema de mensajería entre un nodo dentro de la celda y el nodo que deja la celda, el cual funciona en el siguiente orden: (1) Primero el nodo que deja la celda, avisa al nodo dentro de la celda de la cual salió sobre su salida y entrega el listado de datos en la caché que puede entregar, los cuales solo corresponden a los POI ubicados dentro de la celda. (2) El nodo de dentro de la celda revisa estos datos y envía un aviso de dato repetido si encuentra algún dato que ya está en su caché, en el caso de no haber datos repetidos, se pasa directamente a la etapa (4) del esquema de mensajería. (3) El nodo que deja la celda entrega una lista actualizada de datos a entregar, además en el caso de existir un dato repetido, el nodo que deja la celda elimina este dato dentro de la caché para evitar la redundancia. (4) El nodo dentro de la celda envía su disponibilidad de espacio, para cerciorarse si es posible proceder a la entrega de datos. (5) El nodo que deja la celda envía los datos de acuerdo a la disponibilidad de espacio del nodo dentro de la celda. (6) El nodo dentro de la celda envía un mensaje confirmando la recepción del mensaje anterior.

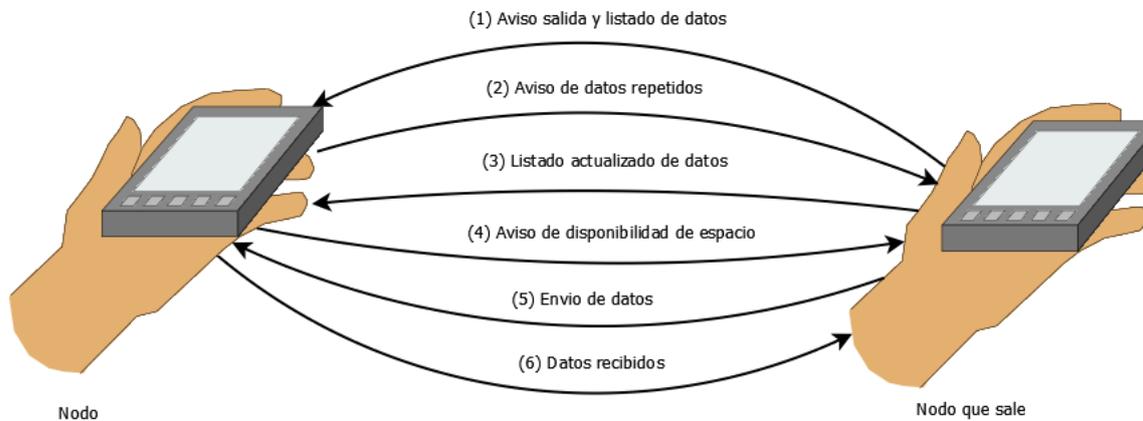


Figura 5.4: Protocolo de Salida 1

5.3.3.2 Protocolo de Salida 2

El protocolo numero 2 funciona como se aprecia se aprecia en la Figura 5.5: (1) El nodo que deja la celda avisa al nodo dentro de la celda de su salida. (2) El nodo dentro de la celda envía al nodo que deja la celda su disponibilidad de espacio, en el caso de no tener más espacio, el protocolo de mensajería termina. (3) El nodo que deja la celda envía el listado de datos que puede entregar. (4) El nodo dentro de la celda avisa de datos repetidos. (5) El nodo que deja la celda entrega nuevos datos. (6) El nodo dentro de la celda envía un mensaje confirmando de recepción de los datos.

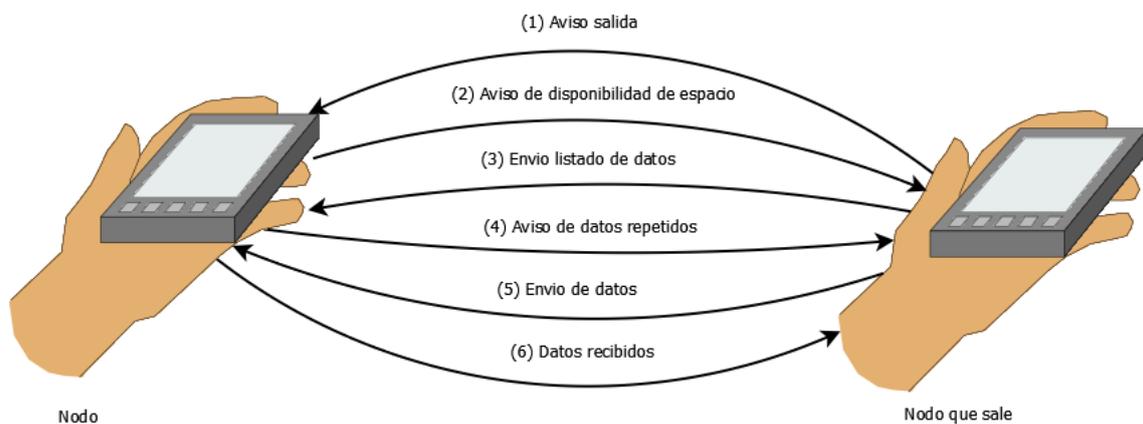


Figura 5.5: Protocolo de Salida 2

5.3.3.3 Diferencias entre protocolos

Se debe destacar que existen diferencias importantes entre estos dos protocolos al cambiar el orden de las acciones a realizar. En protocolo 1 primero se revisa los datos repetidos y

luego se procede a analizar la disponibilidad para realizar la posible entrega de datos. Esto hace que se evite primero la redundancia en el caso de encontrar datos repetidos, pero a la vez puede aumentar el número de mensajes, ya que luego del análisis de redundancia, se procede a revisar la disponibilidad de datos. En cambio en el protocolo 2 al revisar la disponibilidad primero reduce el costo de comunicación (en el caso de que por ejemplo no exista disponibilidad, el protocolo termina y no hay más intercambio de mensajes), pero se puede obtener una redundancia aun mayor porque pueden existir casos en donde existan datos repetidos que no se revisaron ya que no existe una disponibilidad para ingresar más datos y el protocolo termina en ese momento.

5.4 Sistemas propuestos

En base a lo expuesto en la subsección anterior se presenta un sistema individual y se proponen dos sistemas de caching colaborativo, los cuales solo se diferencian en sus protocolos de salida, el resto del funcionamiento es el mismo para los tres.

5.4.1 Sistema Individual (SI)

En este sistema, durante su protocolo de salida, no se entregan datos, es decir, los usuarios se quedan con todos sus datos y se redistribuirán de acuerdo a una estrategia de gestión de caché elegida previamente (en el caso de que no quede espacio en la memoria caché).

5.4.2 Sistema Colaborativo 1 (SC1)

En este sistema, se utiliza el Protocolo de Salida 1, para la entrega de datos. Como en el sistema anterior, los datos se distribuyen de acuerdo a una estrategia de gestión de caché elegida previamente.

5.4.3 Sistema Colaborativo 2 (SC2)

Se usa el Protocolo de Salida 2, para la entrega de datos. También los datos se distribuyen de acuerdo a una estrategia de gestión de caché elegida previamente.

Capítulo 6

Resultados y Discusión

6.1 Configuración de ciudad simulada

En la Sección 5.2, se menciona que la ciudad está conformada por celdas, cada una de estas celdas tiene un tamaño de 10x10 pasos. Esta unidad de medida llamada “pasos”, pasará a ser llamada simplemente “p”. La ciudad es diseñada para ser de un tamaño de 12x12 celdas. Para medir la velocidad de los nodos se usa pasos/segundo, es decir (p/s). Cuando trabaja en medidas de distancias se utiliza unidades. Para el Radio de conexión a la hora de generar una consulta por rango, el R mínimo a usar es de 14 p, ya que el radio debe ser mínimo equivalente a la diagonal de la celda para alcanzar a generar una circunferencia que cubra toda la celda. La diagonal para una celda de 10 p x10 p sería 14.14 p, lo cual se puede redondear a 14 p. El tamaño de la caché corresponde al número de POIs que el nodo puede almacenar hasta llenarse, si se dice tamaño 8, quiere decir que un nodo puede almacenar como máximo 8 POIs, cuando intente almacenar el POI número 9, se procederá a hacer uso de un sistema de gestión de caché predeterminado. Las variables escogidas para los experimentos son las de:

- Tamaño Ciudad = $12 \times 12 = 144$ celdas.
- Tamaño celda = 10×10 p.
- Tiempo simulación = 300 (s).
- Nodos = 400.
- Tamaño caché = 4.
- Velocidad = 4 (p/s).

6.1.1 Elección de gestión de caché

Antes de realizar los experimentos principales, se realizaron experimentos previos para definir cuál es el método de elección de gestión de caches. La configuración escogida corresponde lo descrito anteriormente. Los experimentos se realizaron usando un sistema individual de consulta por celda.

Como se puede apreciar en la Tabla 6.1, los mejores estrategias de gestión de caché, son la de eliminación por antigüedad y la de eliminación por distancia, las cuales producen una menor cantidad de consultas al LBS. La estrategia que entrega un mejor resultado es la de la eliminación por distancia, la cual es la que será usada en los experimentos siguientes, a la hora de efectuar una gestión de la caché.

Cuadro 6.1: Tipos de gestión caché y número de respuestas

Tipos gestión	Respuestas Caché	Respuestas Nodos	Accesos al LBS
1. No hay gestión.	5.752	12.289	101.959
2. Eliminar más antiguo.	50.920	53.744	15.336
3. Eliminar al azar.	45.063	55.406	19.531
4. Eliminar el mas distante.	52.205	53.097	14.698
5. Resetear la caché.	24.919	52.661	42.420
6. Eliminar el más popular.	40.120	57.170	22.710
7. Eliminar menos Popular/Distante.	43.937	56.780	19.283
8. Reseteo caché al entrar en celda.	39.008	55.868	25.124
9. Eliminar al azar.	38.504	60.038	21.458
10. Eliminar más Popular/Distante.	23.367	52.218	44.415

6.2 Resultados experimentales

Se realizaron experimentos cuya variables dependientes son el tamaño de la ciudad, rango de conexión, tiempo de simulación, tamaño de la celda:

- Tamaño Ciudad = $12 \times 12 = 144$ celdas.
- Tamaño celda = 10×10 p.
- Tiempo simulación = 300 (s).

- Radio de conexión= 14 p.
- Radio Máximo = 3 veces Radio de Conexión
- Nodos = 400.
- Tamaño caché = 4.
- Velocidad=4 (p/s).

Las variables dependientes, son la de Tamaño de caché, Velocidad de los Nodos, y número de Nodos. Cada fila de las tablas corresponden a un resultado experimental manteniendo una de estas variables. Se realizaron distintos experimentos analizando distintas variedades de número de nodos, tamaños de caché y velocidades. Las cantidades mostradas a continuación hacen referencia de cantidades promedio, efectuadas en los experimentos, ver Anexo para ver estudios con mayores cantidades de nodos, tamaños caché y velocidades. Para efectos prácticos, se procede a nombrar a al Sistema Individual, Sistema Colaborativo 1 y Sistema Colaborativo 2, por sus abreviaturas SI, SC1 y SC2.

6.2.1 Experimentos con variación de Caché

Para este experimento se trabajan con 400 Nodos y velocidad de 4 (p/s). Los experimentos se dividen en dos tipos, para sistema que utilizan la consulta en celda y sistema que utilizan la consulta en base a un rango.

6.2.1.1 Sistemas utilizando consulta por celda

A continuación se presentan las tablas de resultados:

Cuadro 6.2: Tamaños caché, Sistema individual por celda

Tamaño Caché	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
2	29.194	57.597	33.209	0,76	24%	3,31
3	19.456	56.471	44.073	0,84	27%	2,58
4	14.398	53.432	52.170	0,88	25%	2,16
5	12.374	50.543	57.083	0,90	26%	1,94
6	10.459	47.676	61.865	0,91	24%	1,78

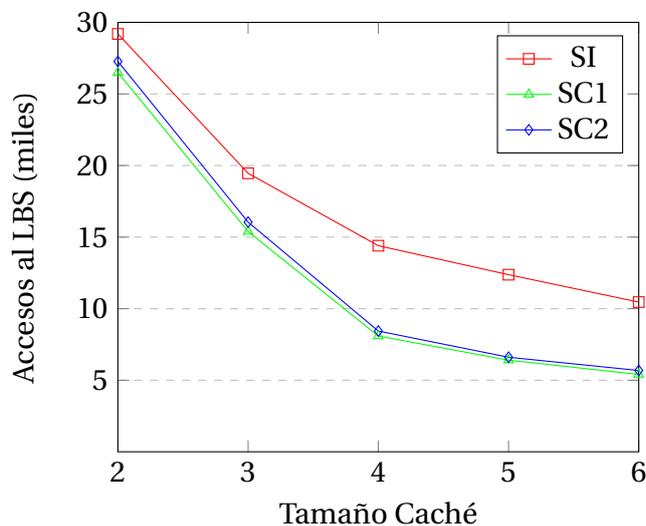
Cuadro 6.3: Tamaños caché, Sistema colaborativo 1 por celda

Tamaño Caché	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
2	26.495	67.879	25.626	0,78	4%	4,62
3	15.387	71.174	33.439	0,87	8%	4,03
4	8.087	72.874	39.039	0,93	9%	3,69
5	6.401	72.331	41.268	0,95	10%	3,63
6	5.404	71.753	42.843	0,95	9%	3,56

Cuadro 6.4: Tamaños caché, Sistema colaborativo 2 por celda

Tamaño Caché	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
2	27.270	66.850	25.880	0,77	6%	4,68
3	16.046	69.619	34.335	0,87	10%	4,17
4	8.436	70.981	40.583	0,93	10%	3,85
5	6.606	69.668	43.726	0,94	11%	3,74
6	5.681	68.725	45.594	0,95	12%	3,75

6.2.1-a Comparación Sistemas por Celda (Accesos al LBS v/s Tamaño Caché)



Se puede observar en el gráfico 6.2.1-a, que a mayor tamaño de caché mejor es el rendimiento, es decir se reduce el número de consultas al LBS, y por consiguiente aumenta la tasa de aciertos, ya que los nodos pueden almacenar más datos. Ambos sistemas colaborativos ofrecen un menor número de consultas al LBS que el sistema individual, a la vez que presentan entre ellos una cantidad de consultas al LBS similares. Al apreciar las Tablas 6.2, 6.3,

6.4, con respecto a la redundancia promedio por celda, va a ir aumentando a medida que se aumenta el tamaño de la caché, debido a que hay más datos en las caches de los nodos, hay mayor probabilidad de que estos estén repetidos. La redundancia presenta una diferencia considerable la entre el sistema individual (siendo la peor), con los sistemas colaborativos (los cuales tienen protocolos de entrega de datos), y también se puede apreciar una diferencia entre el SC1 y el SC2, presentando una menor redundancia el SC1, lo cual valida el sistema de revisión de repetidos previo al listado de datos que efectúa este sistema, a diferencia del SC2. El costo de comunicación en todos los sistemas va a ir disminuyendo a medida que aumenta la caché, debido a que la posibilidad de encontrar un valor dentro de la caché del nodo va a ir aumentando. En este caso el sistema individual, genera un impacto mayor en la reducción del costo de comunicación si se compara con los otros sistemas colaborativos, debido a que, al haber una mayor redundancia es más probable que la respuesta se encuentre más rápido por lo que el número de mensajes se verá reducido, en comparación con los sistemas colaborativos, en los cuales es más probable que se consulte a más usuarios para conocer la respuesta.

6.2.1.2 Sistemas utilizando consulta por rango

A continuación se presentan las tablas de resultados:

Cuadro 6.5: Tamaños caché, Sistema individual por rango

Tamaño Caché	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
2	21.949	64.697	33.354	0,82	24 %	70,59
3	9.852	65.932	44.216	0,92	26 %	33,38
4	4.804	63.035	52.161	0,96	25 %	17,51
5	2.562	59.646	57.792	0,98	27 %	10,55
6	1.512	57.365	61.123	0,99	27 %	6,92

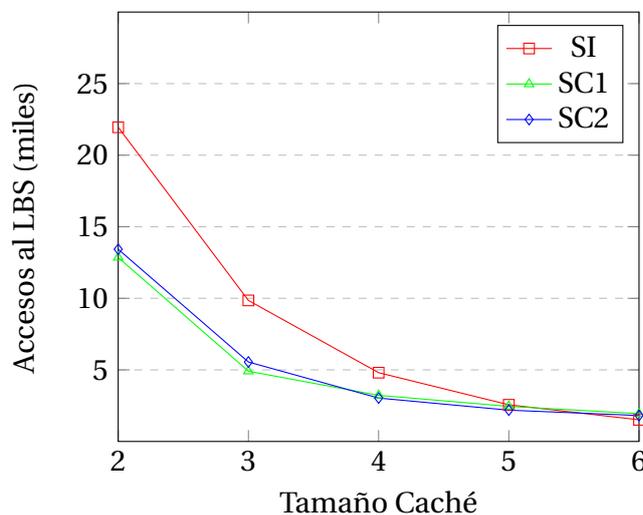
Cuadro 6.6: Tamaños caché, Sistema colaborativo 1 por rango

Tamaño Caché	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
2	12.854	94.486	12.660	0,89	4%	70,08
3	4.908	105.245	9.847	0,96	7%	77,69
4	3.223	107.383	9.394	0,97	6%	74,85
5	2.452	107.976	9.572	0,98	9%	71,94
6	1.943	108.277	9.780	0,98	11%	70,09

Cuadro 6.7: Tamaños caché, Sistema colaborativo 2 por rango

Tamaño Caché	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
2	13.428	84.829	21.743	0,89	6%	60,53
3	5.551	89.394	25.055	0,95	9%	44,85
4	3.031	90.733	26.236	0,97	10%	38,09
5	2.186	91.408	26.406	0,98	10%	38,35
6	1.812	92.280	25.908	0,98	12%	36,97

6.2.1-b Comparación Sistemas por Rango (Accesos al LBS v/s Tamaño Caché)



Al trabajar con rangos, si bien los resultados de los sistemas colaborativos son mejores con una mejor cantidad de consultas al LBS, como se aprecia en el gráfico 6.2.1-b, al aumentar el tamaño de la caché los tres sistemas tienden a entregar un valor similar, con una tasa de acierto casi idéntica con una cache de tamaño 6. En comparación con una consulta por celda presentada en las tablas anteriores, el número de consultas al LBS es resulta menor,

esto ocurre debido a que una consulta por rango consulta a nodos vecinos fuera de la celda, entregando una mayor probabilidad de respuesta sin la necesidad de ir al LBS. Como se puede ver en las Tablas 6.5, 6.6, 6.7, este tipo de consulta también afecta al número de mensajes, la cual si bien se ve reducida al aumentar la caché, es más de 10 veces mayor que al de los sistemas de consulta por celda. Cabe destacar que en este ámbito, el sistema individual es el que posee un menor costo de comunicación y más se ve beneficiada al aumentar la caché, debido que al mantener y no entregar datos, estos datos pueden servir para ayudar nodos de celdas vecinas cuando consultan, gracias al alcance de los rangos, lo que reduce el costo de comunicación, ya que mientras no se encuentre la respuesta se sigue consultando y ampliando los rangos hasta lo que indique el radio máximo, esto se traduce en un mayor costo de comunicación, que es lo que ocurre en los casos de los sistemas colaborativos. En el ámbito de la redundancia los sistemas colaborativos presentan una menor redundancia que el sistema individual, debido a sus protocolos de entrega de datos.

6.2.2 Experimentos con variación de velocidades

Para este experimento se trabajan con 400 Nodos y tamaño de caché de 4. Los experimentos se dividen en dos tipos, para sistema que utilizan la consulta en celda y sistema que utilizan la consulta en base a un rango.

6.2.2.1 Sistemas utilizando consulta por celda

A continuación se presentan las tablas de resultados:

Cuadro 6.8: Velocidad, Sistema individual por celda

Velocidad	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
2	6.483	37.330	76.187	0,95	34 %	1,2
3	10.461	46.659	62.880	0,91	32 %	1,69
4	14.398	53.432	52.170	0,88	25 %	2,16
5	19.485	57.602	42.913	0,84	23 %	2,59
6	23.148	60.402	36.450	0,81	21 %	2,93

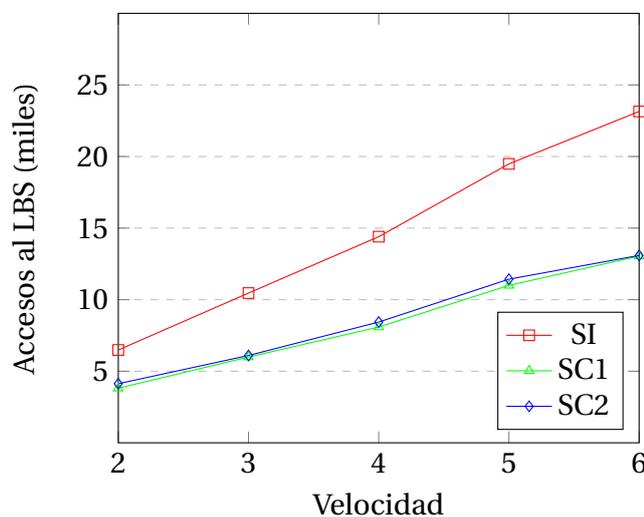
Cuadro 6.9: Velocidad, Sistema colaborativo 1 por celda

Velocidad	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
2	3.814	70.982	45.204	0,97	10%	2,87
3	5.989	71.648	42.363	0,95	9%	3,28
4	8.087	72.874	39.039	0,93	9%	3,69
5	10.997	74.495	34.508	0,91	9%	4,25
6	13.043	75.849	31.108	0,89	8%	4,67

Cuadro 6.10: Velocidad, Sistema colaborativo 2 por celda

Velocidad	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
2	4.128	66.111	49.761	0,97	14%	2,77
3	6.100	68.942	44.958	0,95	13%	3,34
4	8.436	70.981	40.583	0,93	10%	3,85
5	11.430	72.786	35.784	0,90	9%	4,42
6	13.092	74.747	32.161	0,89	8%	4,91

6.2.2-a Comparación Sistemas por Celda (Accesos al LBS v/s Velocidad)



Cuando un nodo se mueve a menor velocidad, su permanecía en la celda va a ser mayor, por lo que la probabilidad de que el valor consultado se encuentre en la caché de algún usuario dentro de la celda va a ser mayor y va a ir disminuyendo a medida que aumente la velocidad, por consiguiente aumentan las consultas al LBS cuando aumenta la velocidad. Este aumento sucede en todos los sistemas, pero son mayores las consultas al LBS por

parte del sistema individual que en el caso de los sistemas colaborativos, como se puede observar en el gráfico 6.2.2-a. Los sistemas colaborativos al realizar una entrega de datos cuando abandonan la celda, la información de los POIs se mantiene más tiempo en la celda, reduciendo el impacto que genera el aumento de velocidad (el cual le afecta a ambos sistemas, en cuanto a consultas al LBS). Al observar las tablas 6.8, 6.9, 6.10, la redundancia se ve reducida en todos los sistemas a medida que aumenta la velocidad debido que la mayoría de los nodos en cada segundo van a pasar a una celda nueva, recibiendo nueva información y la repetición de los datos va a verse reducida ya que va a ser más común que se encuentren nodos que lleguen por primera vez a la celda. Se observa, que los sistemas colaborativos al hacer uso de una entrega de datos, presentan una redundancia menor que los sistemas individuales. El costo de comunicación en el sistema individual es menor que en los sistemas colaborativos, ya que al no generarse una entrega de datos, se reducen también los mensajes generados.

6.2.2.2 Sistemas utilizando consulta por rango

A continuación se presentan las tablas de resultados:

Cuadro 6.11: Velocidad, Sistema individual por rango

Velocidad	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
2	3.155	40.892	75.953	0,97	35 %	10,56
3	4.132	52.638	63.230	0,97	29 %	14,66
4	4.804	63.035	52.161	0,96	25 %	17,51
5	5.378	71.713	42.909	0,96	23 %	20,65
6	5.678	78.018	36.304	0,95	20 %	22,45

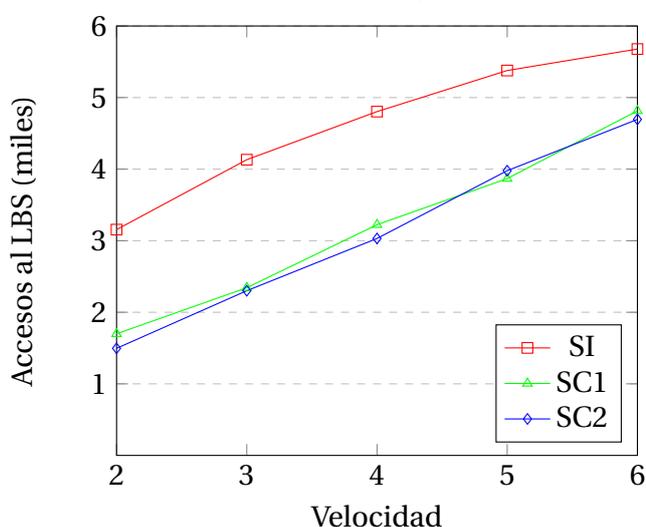
Cuadro 6.12: Velocidad, Sistema colaborativo 1 por rango

Velocidad	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
2	1.699	105.710	12.591	0,99	9 %	58,14
3	2.343	106.726	10.931	0,98	7 %	68,94
4	3.223	107.383	9.394	0,97	6 %	74,85
5	3.868	107.990	8.142	0,97	6 %	83,61
6	4.815	107.374	7.811	0,96	6 %	89,41

Cuadro 6.13: Velocidad, Sistema colaborativo 2 por rango

Velocidad	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
2	1.496	84.638	33.866	0,99	12%	22,96
3	2.301	87.843	29.856	0,98	11%	33,15
4	3.031	90.733	26.236	0,97	10%	38,09
5	3.978	92.369	23.653	0,97	8%	45,7
6	4.695	93.721	21.584	0,96	9%	51,58

6.2.2-b Comparación Sistemas por Rango (Accesos al LBS v/s Velocidad)



Se aprecia en el gráfico 6.2.2-b que ambos sistema colaborativos presentan un menor número de consultas al LBS que el sistema individual. En todos los sistemas también el número de consultas al LBS, va en aumento a medida que aumenta la velocidad. Al generarse consultas por un rango, es decir se consulta a sus vecinos, todos los sistemas presentan una menor cantidad de consultas al LBS que sus mismas versiones en el caso de consultas por celda. Al observar las Tablas 6.11, 6.12, 6.13, se aprecia el enorme aumento del costo de comunicación en todos los casos en comparación con las consultas por celda, debido a que también se consulta a los vecinos de la celda. El aumento en el costo de comunicación va a ir en aumento a medida que aumenta la velocidad, debido a que cada vez es más difícil entrar una respuesta a una velocidad mayor, ya que, como fue explicado anteriormente, los nodos están en celdas nuevas, con nueva información que almacenarán, lo que hace a la vez que se genere (al igual que en el caso de consulta por celda) una redundancia reducida.

6.2.3 Experimentos con variación de número de nodos

Para este experimento se trabajan con una velocidad de 4 (p/s) y tamaño de caché de 4. Los experimentos se dividen en dos tipos, para sistema que utilizan la consulta en celda y sistema que utilizan la consulta en base a un rango.

6.2.3.1 Sistemas utilizando consulta por celda

A continuación se presentan las tablas de resultados:

Cuadro 6.14: Número de Nodos, Sistema individual por celda

Número Nodos	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
200	15.973	18.164	25.863	0,73	11%	1,87
300	15.986	34.448	39.566	0,82	17%	2,03
400	14.398	53.432	52.170	0,88	25%	2,16
500	12.421	72.067	65.512	0,92	31%	2,22
600	10.366	91.666	77.968	0,94	35%	2,25

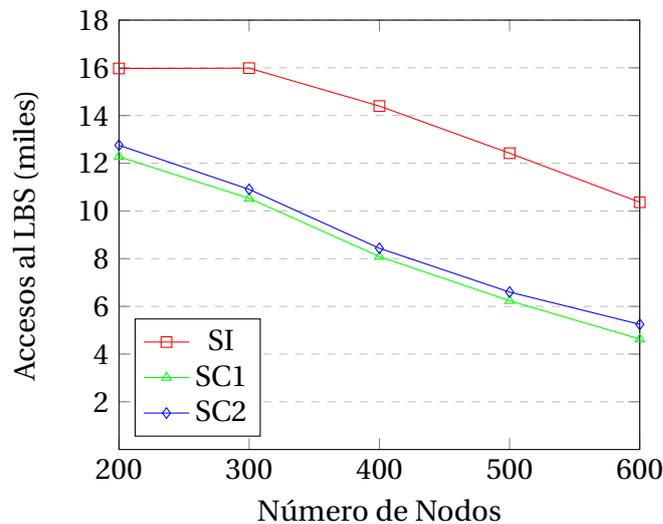
Cuadro 6.15: Número de nodos, Sistema colaborativo 1 por celda

Número Nodos	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
200	12.275	24.297	23.428	0,80	2%	3,37
300	10.531	47.542	31.927	0,88	6%	3,58
400	8.087	72.874	39.039	0,93	9%	3,69
500	6.235	100.010	43.755	0,96	11%	3,90
600	4.628	127.945	47.427	0,97	13%	4,10

Cuadro 6.16: Número de Nodos, Sistema colaborativo 2 por celda

Número Nodos	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
200	12.755	23.977	23.268	0,79	4%	3,53
300	10.905	46.392	32.703	0,88	8%	3,70
400	8.436	70.981	40.583	0,93	10%	3,85
500	6.600	96.851	46.549	0,96	12%	3,94
600	5.246	122.920	51.834	0,97	15%	4,03

6.2.3-a Comparación Sistemas por Celda (Accesos al LBS v/s Número de Nodos)



Se puede observar en el gráfico 6.2.3-a que cuando existe un aumento de nodos, se va a ir reduciendo el número de consultas al LBS y se va a ir aumentando la tasa de aciertos, ya que es más probable que exista un nodo que posea la respuesta a la consulta. En este sistema de consultas por celda, los sistemas colaborativos presentan una menor cantidad de consultas al LBS que el sistema individual, debido a que mientras más usuarios existan es más difícil que una celda quede vacía en cualquier momento de la simulación. Si una celda queda vacía el sistema de entrega de datos pierde su efectividad ya que no hay nodos a quien entregar estos datos. En cuanto a la cantidad de consultas al LBS, se ve que existe una constante de que el sistema SC1 produce siempre ligeramente menos consultas al LBS que el SC2. Se puede observar en las Tablas 6.14, 6.15, 6.16, que a medida que existan más nodos la redundancia promedio en celda va a ser mayor, debido a que existen más datos en la celda. Se puede observar que la redundancia también aumenta cuando hablamos de los sistemas colaborativo, pero en un nivel mucho menor. Incluso se puede observar que el SC1 presenta una menor redundancia que el SC2. Esta menor redundancia de los sistemas colaborativo, es debido a que sus protocolos de entrega de datos ayudan a mitigar la generación de datos repetidos. El costo de comunicación no se ve afectado tan fuertemente para el caso del sistema individual, ya que este solo se aplica en la consulta que efectúa, el cual es por celda, por lo que al aumentar la cantidad de nodos, combinado con el factor de que si encuentra un valor, no se sigue consultando, el costo de comunicación generado no se ve tan afectado. Por el contrario en los casos de los sistemas colaborativos, el costo de comunicación generado es mayor debido a la cantidad de mensajes generados en los protocolos de entrega de datos, también se aprecia que el aumento de este costo de comunicación a medida que aumenta los nodos no es significativo, debido que es esto funciona a nivel de celda, no a nivel rangos.

6.2.3.2 Sistemas utilizando consulta por rango

A continuación se presentan las tablas de resultados:

Cuadro 6.17: Número de Nodos, Sistema individual por rango

Número Nodos	Accesos al LBS	Respuestas Nodos	Respuestas Cache	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
200	8.780	25.159	26.061	0,85	13%	30,10
300	6.740	44.206	39.054	0,93	20%	23,60
400	4.804	63.035	52.161	0,96	25%	17,51
500	3.114	81.168	65.718	0,98	31%	12,15
600	2.123	99.822	78.055	0,99	36%	8,90

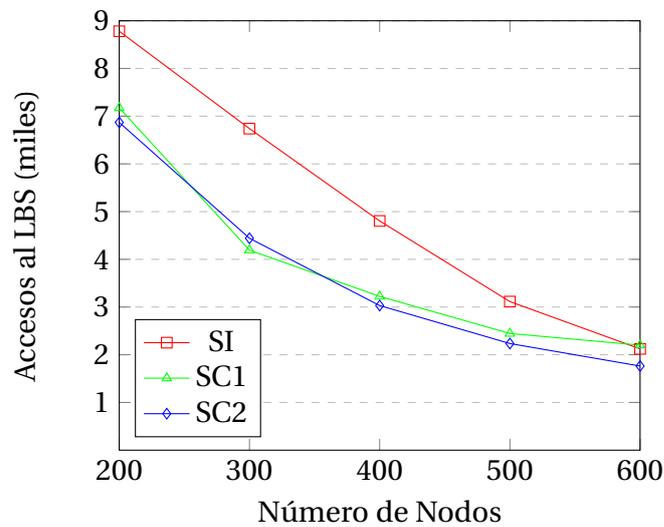
Cuadro 6.18: Número de Nodos, Sistema colaborativo 1 por rango

Número Nodos	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
200	7.174	41.319	11.507	0,88	5%	43,46
300	4.192	75.898	9.910	0,95	8%	58,79
400	3.223	107.383	9.394	0,97	6%	74,85
500	2.445	138.031	9.524	0,98	7%	88,03
600	2.201	168.434	9.365	0,99	9%	99,40

Cuadro 6.19: Número de Nodos, Sistema colaborativo 2 por rango

Número Nodos	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
200	6.869	34.094	19.037	0,89	5%	30,56
300	4.441	62.356	23.203	0,95	8%	32,48
400	3.031	90.733	26.236	0,97	10%	38,09
500	2.235	118.683	29.082	0,99	11%	45,69
600	1.763	148.049	30.188	0,99	13%	58,00

6.2.3-b Comparación Sistemas por Rango (Accesos al LBS v/s Número de Nodos)



Se aprecia en el gráfico 6.2.3-b, que las consultas al LBS disminuyen a medida que aumenta el número de nodos, lo igual que el experimento anterior, los sistemas colaborativos presentan menos consultas al LBS y una mayor tasa de aciertos que el sistema individual, pero también se puede observar que además de reducirse las consultas generadas al LBS, se empieza a formar una tendencia a la convergencia en cuanto a la cantidad de consultas al LBS, al aumentar la cantidad de nodos en el sistema. Al observar las Tablas 6.17, 6.18, 6.19, la redundancia en los sistemas colaborativos es menor al sistema individual, y va a ir en aumento cuando se incrementa la cantidad de nodos. El sistema SC1 presenta una menor redundancia que el SC2, al igual que en los casos de consulta por celda. En el costo de comunicación se puede apreciar un cambio con respecto a los casos de sistemas por celda. Los sistemas colaborativos aumentan su costo de comunicación a medida que aumenta el número de nodos, en cambio el sistema individual lo disminuye. Esto puede suceder debido a que en la consulta por rango se consulta una mayor cantidad de nodos incluyendo a los vecinos de las celdas, aumentando la cantidad de consultas por cada celda extra consultada. Además para los sistemas colaborativos, al momento de generar un protocolo de entrega de datos se produce una mayor mensajería, al igual como sucedía en las consultas por celda. En el sistema individual, al no poseer estos protocolos, no se produce esta mensajería extra.

6.2.4 Comparación general de sistemas respecto a resultados experimentales

En los casos presentados, se observa que los sistemas colaborativos presentan una menor cantidad de consultas al LBS, que lo presentado por un sistema individual. También se puede

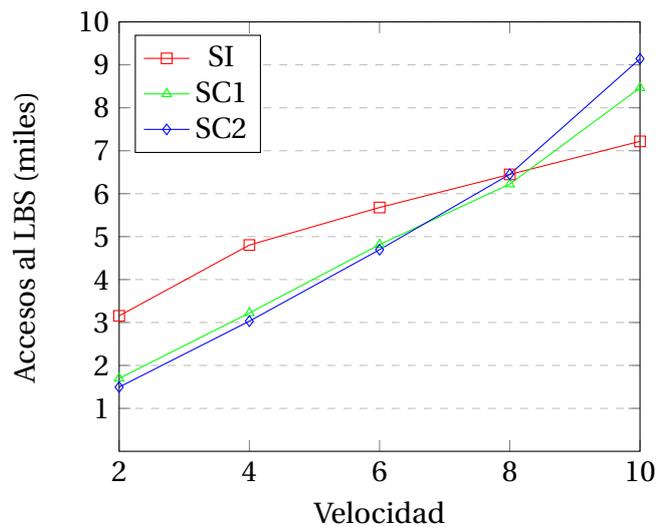
observar que el sistema individual en general tiene un menor costo de comunicación, a la vez que genera una mayor redundancia por celda, esto es debido a la ejecución de los protocolos en los sistemas colaborativos, los cuales ayudan a reducir la redundancia y a la vez aumentan el costo de comunicación.

A través de estos experimentos se ha apreciado que el sistema SC1 presenta una menor redundancia que el SC2, demostrando que la diferencia entre ambos protocolos, la cual radica en el orden de cómo realizar la mensajería, genera un efecto apreciable en los experimentos.

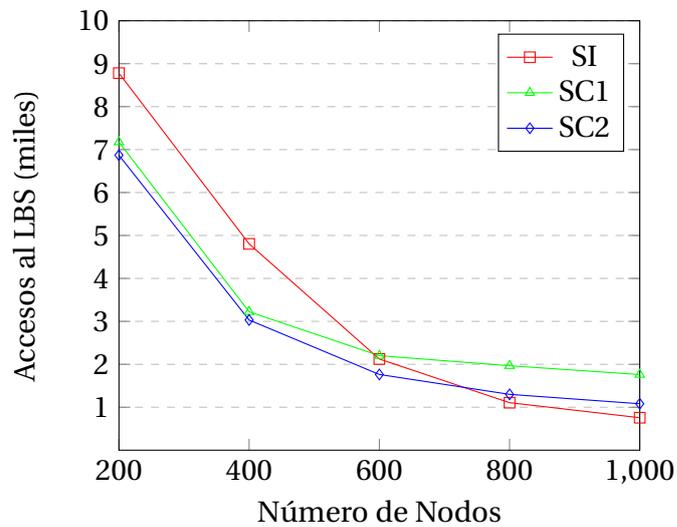
6.2.5 Casos particulares de grandes valores en consultas por rango

A continuación se presentan los gráficos de dos casos de algunas de las variaciones extremas expresadas en las tablas 8.10, 8.11, 8.12 y 8.16, 8.17, 8.18 descritas en el Anexo.

6.2.5-a Comparación Sistemas por Rango (Accesos al LBS v/s Velocidad)



6.2.5-b Comparación Sistemas por Rango (Accesos al LBS v/s Número de Nodos)



Se debe mencionar los casos específicos de sistemas de consultas por rango, con grandes velocidades (Gráfico 6.2.5-a, creado en base a las tablas del Anexo sección 8.1.1) y grandes cantidades de nodos (Gráfico 6.2.5-b, creado en base a las tablas del Anexo sección 8.1.2), se puede observar que cuando la velocidad supera los 8 (p/s) el sistema individual comienza a entregar un mejor resultado, entregando menos consultas al LBS. Al igual cuando se supera la cantidad de 800 nodos en la simulación. El sistema individual al no realizar una entrega de datos y combinado con las consultas por rangos (se pregunta a vecinos que tampoco han entregado sus datos), hace que pueda soportar mejor estas condiciones de grandes velocidades y enormes cantidades de nodos, ya que se pierden menos datos (cabe mencionar que ambos sistemas al realizar una gestión de caché, van a proceder a eliminar datos cuando se requiera agregar nueva información y se posea una caché llena).

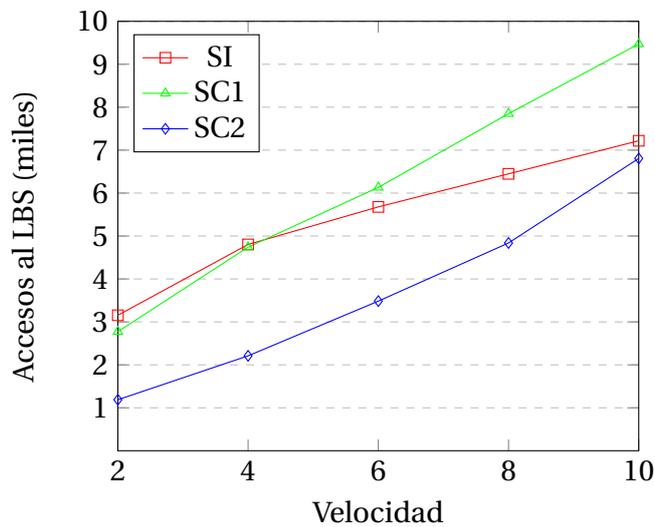
6.2.5.1 Solución en caso de grandes valores en consultas de rango

En las consultas por rangos en el sistema colaborativo, al solo agregar datos nuevos a la caché del nodo en el caso de nadie tenga la respuesta, puede generar dependencia de un nodo que este muy alejado su celda original, cuando este nodo abandone el rango, como ningún vecino tiene la respuesta, se deberá ir al LBS, porque este nodo no va a entregar este dato en cuestión debido a que solo se entregan los datos correspondientes a la celda de salida, no al de una celda antigua.

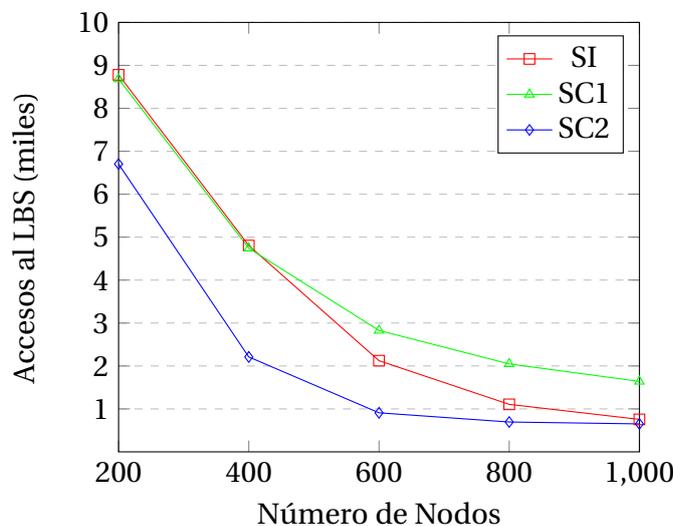
Para los protocolos de SC1 y SC2 se les agrega la opción de que en caso de consultas por rango se encuentre la respuesta en un nodo fuera de la celda, el nodo que consulta almacena

esta información ya que se considera un dato nuevo para todos los nodos dentro de la celda. En base lo anterior y mantenido la misma configuración de los gráficos 6.2.5-a a 6.2.5-b, se obtienen los siguientes resultados:

6.2.5.1-a Comparación Sistemas por Rango (Accesos al LBS v/s Velocidad)



6.2.5.1-b Comparación Sistemas por Rango (Accesos al LBS v/s Número de Nodos)



Se puede observar en los nuevos resultados de los gráficos 6.2.5.1-a y 6.2.5.1-b, el protocolo SC2 presenta mejores resultados ya que elimina menos datos en comparación con SC1. En cambio el SC1, al revisar primero los datos repetidos antes de proceder a revisar la disponibilidad de espacio en la caché, va a eliminar muchos datos que podrían necesitarse a futuro a costa de reducir la redundancia. Con esta solución se puede observar que en ambos casos el protocolo SC2 entrega también mejor resultado que el sistema individual de consulta.

Capítulo 7

Conclusiones y Trabajo futuro

7.1 Conclusiones

La privacidad en servicios basados en ubicación es un factor importante a tener en cuenta y cada día va más en aumento debido a la proliferación de dispositivos móviles con acceso a Internet, siendo la privacidad de ubicación y del contenido de la consulta un factor de riesgo al verse vulnerada al contactar con el LBS.

En la literatura existen redes MANET para evitar el contacto con el LBS las cuales a la hora de realizar consultas, preguntan primero en la red antes de ir al LBS, pero gestionan su memoria de manera individual. Además, no se presentan estudios que utilicen una colaboración de la memoria caché de los nodos sin el uso de repositorios externos de consultas, es decir, no hay estudios que utilicen una colaboración usando exclusivamente la información en la caché de los nodos para responder consultas.

A lo largo de nuestros experimentos comprobamos nuestra hipótesis: *“Un caché colaborativo va a permitir una reducción del acceso al LBS y por lo tanto, una mayor protección de la privacidad de ubicación en comparación con el uso de un caché local, donde cada nodo decide cómo gestionar su propio caché”*. Sobre los puntos de la hipótesis: *“Nodos se coordinan en qué almacenar, sin generar un alto costo de comunicación”* y *“Múltiples nodos aprovechen la respuesta de una consulta”*, el último punto de aprovechar la respuesta de una consulta no se implementó en los experimentos, se deja como trabajo futuro, su implementación.

Con respecto al objetivo general y objetivos específicos, se cumplieron en ambos casos. Para el objetivo general, se evaluó el rendimiento de técnicas de caching colaborativo, para

implementar un servicio LBS en una MANET. En los objetivos específicos, se seleccionaron técnicas de caching colaborativo entre usuarios de una MANET (Objetivo específico 1), se incorporaron técnicas de caché colaborativo en una MANET (Objetivo específico 2) y se evaluó el rendimiento de las técnicas empleadas (Objetivo específico 3). Se debe mencionar que en este último objetivo específico se mencionaba “*evaluar el rendimiento con otras existentes en la literatura*”, pero en este caso, al tratarse de técnicas con una base nueva (el no uso de estructuras externas y a la vez, la métrica de la tasa de aciertos incluye aciertos en caches de vecinos, no exclusivamente en la propia caché como se menciona en la literatura) se evaluó el rendimiento con respecto a otras técnicas también creadas en el presente informe.

Si bien como se menciona anteriormente, el uso de técnicas de caching colaborativo sin el uso de estructuras externas no se encuentra en la literatura, sí se evaluaron las técnicas de gestión la caché individual de cada usuario, las cuales sí existen en la literatura. Con respecto a este tema, comprobamos que la eliminación por distancia es la que entrega mejores resultados a la hora de hacer un buen uso de la memoria, seguida de la eliminación por antigüedad. Cabe destacar que si no existe una gestión de la caché, siempre se presentan los peores resultados.

A través de nuestros experimentos, logramos corroborar en la mayoría de los casos en que se usa un caché colaborativo, se logra una reducción del acceso al LBS y por lo tanto, una mayor protección de la privacidad de ubicación en comparación con el uso de un Sistema Individual donde no se produce una entrega de datos entre los nodos vecinos.

Para la implementación de las técnicas de caching semántico colaborativo, se crearon dos protocolos para realizar una entrega de datos con el uso de un caché colaborativo. En base a estos protocolos se definen dos sistemas de caching colaborativos, siendo el Sistema Colaborativo 1 o SC1 el que entrega mejores resultados, el cual, a diferencia del Sistema Colaborativo 2 o SC2, primero examina datos repetidos antes de proceder a revisar si el nodo es apto para recibir más información (el SC2 realizan esta acción al revés, provocando que exista la posibilidad de que se mantengan mas datos repetidos). Esta diferencia entre los protocolos es observable cuando se trabaja a nivel de consulta por celda, independiente de la velocidad, tamaño de la caché y número de nodos. En un caso promedio de una caché de tamaño 4, velocidad de 4 (p/s) y 400 nodos. El SC1 presenta una mejora del 4% con respecto al SC2 en cuanto a reducción de accesos al LBS. Cuando estos sistemas se comparan con el Sistema Individual, el SC1 presenta una mejora del 43% en reducción de accesos al LBS

con respecto al Sistema Individual, y el SC2 una mejora del 41 %. Se llega a una diferencia de hasta el 55 % del SC1 y 49 % del SC2 en reducción de accesos al LBS, con respecto al Sistema Individual cuando aumentamos el experimento a 600 nodos.

Cuando se trabaja con consultas por rangos, es decir cuando se trabaja con consultas a múltiples celdas, las cuales se van replicando a más celdas. Con los parámetros de caché de tamaño 4, velocidad de 4 (p/s) y 400 nodos, el SC1 presenta una mejora del 32 % en reducción de accesos al LBS con respecto al Sistema Individual, y el SC2 una mejora del 36 %. Solo en los casos de grandes velocidades o de grandes cantidades de nodos, los sistemas colaborativos no presentan mejoras con respecto al Sistema Individual.

Independientemente del tipo de consulta realizada, los sistemas colaborativos presentan una menor redundancia por celda que el Sistema Individual. Bajo los parámetros de consulta por celda con caché de tamaño 4, velocidad de 4 (p/s) y 400 nodos, el Sistema individual muestra una redundancia promedio por celda del 25 % y los sistemas colaborativos SC1 y SC2, del 9 % y 10 % respectivamente. Ambos sistemas colaborativos también presentan un mayor costo de comunicación (bajo los parámetros anteriormente mencionados), el SC1 y SC2 aumentan su costo de comunicación al 41 % y 43 % en comparación al Sistema Individual, es decir presentan una mayor mensajería.

7.2 Trabajo futuro

El modelo de movimiento de la ciudad simulada en el presente informe es el de random walk, si se quiere aplicar este modelo a un escenario realista, se debería cambiar a un modelo de movimiento más “humano”. Un modelo de rutas podría ser la solución ideal para realizar un modelo más realista. En este tipo de modelos los nodos siguen una ruta predefinida, siendo lo más sencillo desde un sector del mapa hacia otro sector, pasando por distintos puntos de interés, para luego volver a su posición original, emulando un camino o ruta de una persona desde su casa al trabajo y viceversa. A este modelo, se le podría añadir la gestión de caché basada en el historial de lugares visitados en la ruta del nodo, y así seleccionar un orden de prioridad a la hora de eliminar datos por insuficiencia de espacio en la caché.

Para la creación de una simulación de una ciudad más realista, se podrían redistribuir los puntos de interés de manera más cercana a una ciudad real, ya que en el sistema actual, la localización de los puntos de interés es aleatoria. En una ciudad es común que existan zonas con determinados puntos de interés, como el caso de restaurantes y hoteles, al igual que existen zonas con ausencia de puntos de interés como el caso de parques, ríos y autopistas. A esta ciudad también se le podrían añadir estructuras que funcionen a modo de antenas que trabajen de intermediario para mejorar la conexión entre los nodos de la simulación.

En el sistema creado, el intercambio de información está basado en la confianza entre los usuarios, por lo que podría verse afectado por ataques a la MANET por entidades externas a la red de usuarios. Se deja como posible trabajo futuro la implementación de mecanismos que eviten o disipen el ataque a la red MANET, como lo es el uso de mensajes encriptados entre los nodos en la red (27) y (28).

En el sistema no se implementó la l-Diversidad ni el k-Anonimato, se asume que un usuario cuando contacta al LBS, lo hace luego que aplicar estas técnicas, queda como un posible trabajo futuro la implementación de estas técnicas en la simulación. También en este sistema se deja como trabajo futuro la optimización de que múltiples nodos aprovechen la respuesta de una consulta, además del uso del concepto de “similaridad semántica”, con el cual se podrían añadir puntos de interés de temáticas similares como una respuesta correcta.

Bibliografía

- [1] M. S. Alrahhah, M. Khemakhem, and K. Jambi, “A survey on privacy of location-based services: Classification, inference attacks, and challenges,” *Journal of Theoretical and Applied Information Technology*, vol. 95, no. 24, pp. 6719–6740, 2017.
- [2] B. Niu, Q. Li, X. Zhu, G. Cao, and H. Li, “Achieving k-anonymity in privacy-aware location-based services,” *Proceedings - IEEE INFOCOM*, pp. 754–762, 2014.
- [3] B. Niu, X. Zhu, W. Li, H. Li, Y. Wang, and Z. Lu, “A personalized two-tier cloaking scheme for privacy-aware location-based services,” *2015 International Conference on Computing, Networking and Communications, ICNC 2015*, no. D1d, pp. 94–98, 2015.
- [4] C. Y. Chow, M. F. Mokbel, and W. G. Aref, “Casper*: Query processing for location services without compromising privacy,” *ACM Transactions on Database Systems*, vol. 34, no. 4, 2009.
- [5] M. Gruteser and D. Grunwald, “Anonymous usage of location-based services through spatial and temporal cloaking,” *Proceedings of the 1st International Conference on Mobile Systems, Applications and Services, MobiSys 2003*, pp. 31–42, 2003.
- [6] B. Niu, Q. Li, X. Zhu, G. Cao, and H. Li, “Enhancing privacy through caching in location-based services,” *Proceedings - IEEE INFOCOM*, vol. 26, pp. 1017–1025, 2015.
- [7] O. Temuujin, J. Ahn, and D. H. Im, “Efficient l-diversity algorithm for preserving privacy of dynamically published datasets,” *IEEE Access*, vol. 7, pp. 122878–122888, 2019.
- [8] S. Dar, M. Franklin, B. Jonsson, D. Srivastava, and M. Tan, “Semantic Data Caching and Replacement,” *Proceedings of the 22th International Conference on Very Large Data Bases*, 1996.
- [9] G. Roşu and T. F. Şerbănuţă, “An overview of the K semantic framework,” *Journal of Logic and Algebraic Programming*, vol. 79, no. 6, pp. 397–434, 2010.

-
- [10] Z. Xiang and B. Pan, "Travel queries on cities in the United States: Implications for search engine marketing for tourist destinations," *Tourism Management*, vol. 32, no. 1, pp. 88–97, 2011.
- [11] W. Li, B. Niu, J. Cao, Y. Luo, and H. Li, "A personalized range-sensitive privacy-preserving scheme in LBSs," *Concurrency and Computation: Practice and Experience*, vol. 32, no. 5, pp. 1–11, 2020.
- [12] K. Jung and S. Park, "Collaborative caching techniques for privacy-preserving location-based services in peer-to-peer environments," *Proceedings - 2017 IEEE International Conference on Big Data, Big Data 2017*, vol. 2018-Janua, pp. 4497–4506, 2017.
- [13] S. Zhang, Q. Liu, and G. Wang, "A caching-based privacy-preserving scheme for continuous location-based services," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 10067 LNCS, pp. 73–82, 2016.
- [14] T. Peng, Q. Liu, D. Meng, and G. Wang, "Collaborative trajectory privacy preserving scheme in location-based services," *Information Sciences*, vol. 387, pp. 165–179, 2017.
- [15] S. Zhang, X. Li, Z. Tan, T. Peng, and G. Wang, "A caching and spatial [Formula presented]-anonymity driven privacy enhancement scheme in continuous location-based services," *Future Generation Computer Systems*, vol. 94, pp. 40–50, 2019.
- [16] H. N. S. S. Jagarlapudi, S. Lim, J. Chae, G. S. Choi, and C. Pu, "Drone Helps Privacy: Sky Caching Assisted k-Anonymity in Spatial Querying," *IEEE Systems Journal*, vol. 16, no. 4, pp. 6360–6370, 2022.
- [17] Y. Alsaawy, A. Alkhodre, F. A. Eassa, and A. A. A. Sen, "Triple cache approach for preserving privacy and enhancing performance of LBS," *Proceedings of the 2019 6th International Conference on Computing for Sustainable Global Development, INDIACom 2019*, no. March, pp. 1277–1281, 2019.
- [18] X. Mu, H. Shen, and Z. Lu, "A temporal caching-aware dummy selection location Algorithm," *Proceedings - 2019 20th International Conference on Parallel and Distributed Computing, Applications and Technologies, PDCAT 2019*, pp. 501–504, 2019.
- [19] R. Gupta and U. P. Rao, "Achieving location privacy through CAST in location based services," *Journal of Communications and Networks*, vol. 19, no. 3, pp. 239–249, 2017.

-
- [20] H. Jin and P. Papadimitratos, “Resilient privacy protection for location-based services through decentralization,” *Proceedings of the 10th ACM Conference on Security and Privacy in Wireless and Mobile Networks, WiSec 2017*, pp. 253–258, 2017.
- [21] Y. Liu, D. Zhu, and W. Ma, “A novel cooperative caching scheme for Content Centric Mobile Ad Hoc Networks,” *Proceedings - IEEE Symposium on Computers and Communications*, vol. 2016-August, pp. 824–829, 2016.
- [22] J. Sun, W. Xu, C. Wang, and X. Chen, “Response Time-Delay Analysis of Cooperative Cache in MANETs,” *Proceedings of the IEEE International Conference on Software Engineering and Service Sciences, ICSESS*, vol. 2018-November, pp. 637–643, 2019.
- [23] M. Ahmed Elfaki, H. Ibrahim, A. Mamat, M. Othman, and H. Safa, “Collaborative caching priority for processing requests in MANETs,” *Journal of Network and Computer Applications*, vol. 40, no. 1, pp. 85–96, 2014.
- [24] B. D. Shelar and D. K. Chitre, “Use of Hidden Markov Mobility Model for Location Prediction and Biclustering for Cache Replacement in MANET,” vol. 2, no. 7, pp. 292–296, 2015.
- [25] L. V. Bogachev, “Random Walks in Random Environments,” *Encyclopedia of Mathematical Physics: Five-Volume Set*, vol. 4, pp. 353–371, 2004.
- [26] P. Galdames, K. Kim, and Y. Cai, “A generic platform for efficient processing of spatial monitoring queries in mobile peer-to-peer networks,” *Proceedings - IEEE International Conference on Mobile Data Management*, pp. 1–10, 2010.
- [27] I. Bilogrevic, M. Jadliwala, K. Kalkan, J. P. Hubaux, and I. Aad, “Privacy in mobile computing for location-sharing-based services,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 6794 LNCS, pp. 77–96, 2011.
- [28] P. Asuquo, H. Cruickshank, J. Morley, C. P. Ogah, A. Lei, W. Hathal, S. Bao, and Z. Sun, “Security and Privacy in Location-Based Services for Vehicular and Mobile Communications: An Overview, Challenges, and Countermeasures,” *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 4778–4802, 2018.

Capítulo 8

Anexo

8.1 Experimentos con amplias variaciones de Caché

Para este experimento se trabajan con 400 Nodos y velocidad de 4 (p/s). Los experimentos se dividen en dos tipos, para sistema que utilizan la consulta en celda y sistema que utilizan la consulta en base a un rango.

8.1.1 Sistemas utilizando consulta por celda

A continuacion se presentan las tablas de resultados:

Cuadro 8.1: Anexo: Tamaños caché, Sistema individual por celda

Tamaño Caché	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
2	29.194	57.597	33.209	0,76	24 %	3,31
4	14.398	53.432	52.170	0,88	25 %	2,16
8	8.555	44.248	67.197	0,93	24 %	1,52
16	6.118	36.595	77.287	0,95	26 %	1,15
32	4.842	31.839	83.319	0,96	25 %	0,97

Cuadro 8.2: Anexo: Tamaños caché, Sistema colaborativo 1 por celda

Tamaño Caché	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
2	26.495	67.879	25.626	0,78	4 %	4,62
4	8.087	72.874	39.039	0,93	9 %	3,69
8	5.091	69.778	45.131	0,96	11 %	3,46
16	4.373	67.570	48.057	0,96	12 %	3,35
32	4.348	67.264	48.388	0,96	11 %	3,28

Cuadro 8.3: Anexo: Tamaños caché, Sistema colaborativo 2 por celda

Tamaño Caché	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
2	27.270	66.850	25.880	0,77	6 %	4,68
4	8.436	70.981	40.583	0,93	10 %	3,85
8	4.967	66.905	48.128	0,96	11 %	3,71
16	4.323	64.236	51.441	0,96	12 %	3,6
32	4.294	65.429	50.277	0,96	12 %	3,53

8.1.2 Sistemas utilizando consulta por rango

A continuacion se presentan las tablas de resultados:

Cuadro 8.4: Anexo: Tamaños caché, Sistema individual por rango

Tamaño Caché	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
2	21.949	64.697	33.354	0,82	24 %	70,59
4	4.804	63.035	52.161	0,96	25 %	17,51
8	790	52.116	67.094	0,99	24 %	4,2
16	582	41.658	77.760	1,00	25 %	2,77
32	576	35.934	83.490	1,00	24 %	2,48

Cuadro 8.5: Anexo: Tamaños caché, Sistema colaborativo 1 por rango

Tamaño Caché	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
2	12.854	94.486	12.660	0,89	4 %	70,08
4	3.223	107.383	9.394	0,97	6 %	74,85
8	1.531	108.067	10.402	0,99	12 %	63,03
16	933	103.955	15.112	0,99	19 %	71,94
32	642	92.410	26.948	0,99	14 %	70,09

Cuadro 8.6: Anexo: Tamaños caché, Sistema colaborativo 2 por rango

Tamaño Caché	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
2	13.428	84.829	21.743	0,89	6 %	60,53
4	3.031	90.733	26.236	0,97	10 %	38,09
8	1.313	93.587	25.100	0,99	14 %	36,53
16	805	95.083	24.112	0,99	20 %	29,99
32	655	92.117	27.228	0,99	23 %	20,64

8.2 Experimentos con amplias variaciones de velocidades

Para este experimento se trabajan con 400 Nodos y tamaño de caché de 4. Los experimentos se dividen en dos tipos, para sistema que utilizan la consulta en celda y sistema que utilizan la consulta en base a un rango.

8.2.1 Sistemas utilizando consulta por celda

A continuacion se presentan las tablas de resultados:

Cuadro 8.7: Anexo: Velocidad, Sistema individual por celda

Velocidad	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
2	6.483	37.330	76.187	0,95	34%	1,2
4	14.398	53.432	52.170	0,88	25%	2,16
6	23.148	60.402	36.450	0,81	21%	2,93
8	30.926	63.681	25.393	0,74	18%	3,57
10	40.323	63.976	15.701	0,66	13%	4,21

Cuadro 8.8: Anexo: Velocidad, Sistema colaborativo 1 por celda

Velocidad	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
2	3.814	70.982	45.204	0,97	10%	2,87
4	8.087	72.874	39.039	0,93	9%	3,69
6	13.043	75.849	31.108	0,89	8%	4,67
8	17.626	23.392	78.982	0,85	5%	5,67
10	23.326	83.625	13.049	0,81	7%	6,96

Cuadro 8.9: Anexo: Velocidad, Sistema colaborativo 2 por celda

Velocidad	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
2	4.128	66.111	49.761	0,97	14%	2,77
4	8.436	70.981	40.583	0,93	10%	3,85
6	13.092	74.747	32.161	0,89	8%	4,91
8	18.201	78.433	23.366	0,85	9%	6,01
10	23.732	83.275	12.993	0,80	7%	7,32

8.2.2 Sistemas utilizando consulta por rango

A continuacion se presentan las tablas de resultados:

Cuadro 8.10: Anexo: Velocidad, Sistema individual por rango

Velocidad	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
2	3.155	40.892	75.953	0,97	35%	10,56
4	4.804	63.035	52.161	0,96	25%	17,51
6	5.678	78.018	36.304	0,95	20%	22,45
8	6.448	88.327	25.225	0,95	17%	27,57
10	7.218	97.319	15.463	0,94	14%	33,32

Cuadro 8.11: Anexo: Velocidad, Sistema colaborativo 1 por rango

Velocidad	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
2	1.699	105.710	12.591	0,99	9%	58,14
4	3.223	107.383	9.394	0,97	6%	74,85
6	4.815	107.374	7.811	0,96	6%	89,41
8	6.223	107.360	6.417	0,95	6%	94,24
10	8.465	106.797	4.738	0,93	5%	105,25

Cuadro 8.12: Anexo: Velocidad, Sistema colaborativo 2 por rango

Velocidad	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
2	1.496	84.638	33.866	0,99	12%	22,96
4	3.031	90.733	26.236	0,97	10%	38,09
6	4.695	93.721	21.584	0,96	9%	51,58
8	6.451	96.298	17.251	0,95	8%	59,34
10	9.141	99.406	11.453	0,92	6%	70,97

8.3 Experimentos con amplia variación de número de nodos

Para este experimento se trabajan con una velocidad de 4 (p/s) y tamaño de caché de 4. Los experimentos se dividen en dos tipos, para sistema que utilizan la consulta en celda y sistema que utilizan la consulta en base a un rango.

8.3.1 Sistemas utilizando consulta por celda

A continuacion se presentan las tablas de resultados:

Cuadro 8.13: Anexo: Número de Nodos, Sistema individual por celda

Número Nodos	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
200	15.973	18.164	25.863	0,73	11 %	1,87
400	14.398	53.432	52.170	0,88	25 %	2,16
600	10.366	91.666	77.968	0,94	35 %	2,25
800	6.413	129.780	103.807	0,97	44 %	2,29
1000	4.085	165.581	130.334	0,99	52 %	2,26

Cuadro 8.14: Anexo: Número de nodos, Sistema colaborativo 1 por celda

Número Nodos	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
200	12.275	24.297	23.428	0,80	2 %	3,37
400	8.087	72.874	39.039	0,93	9 %	3,69
600	4.628	127.945	47.427	0,97	13 %	4,10
800	2.671	185.079	52.250	0,99	12 %	4,40
1000	1.944	243.080	54.976	0,99	12 %	4,87

Cuadro 8.15: Anexo: Número de Nodos, Sistema colaborativo 2 por celda

Número Nodos	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
200	12.755	23.977	23.268	0,79	4 %	3,53
400	8.436	70.981	40.583	0,93	10 %	3,85
600	5.246	122.920	51.834	0,97	15 %	4,03
800	3.633	176.132	60.235	0,98	16 %	4,13
1000	2.911	230.109	66.980	0,99	17 %	4,17

8.3.2 Sistemas utilizando consulta por rango

A continuación se presentan las tablas de resultados:

Cuadro 8.16: Anexo: Número de Nodos, Sistema individual por rango

Número Nodos	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
200	8.780	25.159	26.061	0,85	13%	30,10
400	4.804	63.035	52.161	0,96	25%	17,51
600	2.123	99.822	78.055	0,99	36%	8,90
800	1.106	135.046	103.848	1,00	46%	5,36
1000	756	168.924	130.320	1,00	51%	3,92

Cuadro 8.17: Anexo: Número de Nodos, Sistema colaborativo 1 por rango

Número Nodos	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
200	7.174	41.319	11.507	0,88	5%	43,46
400	3.223	107.383	9.394	0,97	6%	74,85
600	2.201	168.434	9.365	0,99	6%	99,40
800	1.966	228.182	9.852	0,99	11%	116,33
1000	1.761	288.016	10.223	0,99	13%	123,93

Cuadro 8.18: Anexo: Número de Nodos, Sistema colaborativo 2 por rango

Número Nodos	Accesos al LBS	Respuestas Nodos	Respuestas Caché	Tasa de acierto	Redundancia promedio por celda	Costo de Comunicación
200	6.869	34.094	19.037	0,89	5%	30,56
400	3.031	90.733	26.236	0,97	10%	38,09
600	1.763	148.049	30.188	0,99	13%	58,00
800	1.300	205.232	33.468	0,99	13%	67,53
1000	1.081	263.801	35.118	1,00	13%	78,76