

19



OFICINA ESPAÑOLA DE  
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 568 065**

51 Int. Cl.:

**H04L 12/721** (2013.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **24.02.2014** **E 14156373 (4)**

97 Fecha y número de publicación de la concesión europea: **24.02.2016** **EP 2911349**

54 Título: **Método y sistema para predicción de enlace en computación móvil**

45 Fecha de publicación y mención en BOPI de la traducción de la patente:  
**27.04.2016**

73 Titular/es:

**DEUTSCHE TELEKOM AG (100.0%)**  
**Friedrich-Ebert-Allee 140**  
**53113 Bonn, DE**

72 Inventor/es:

**HUI, PAN;**  
**HUANG, DI y**  
**PEYLO, CHRISTOPH**

74 Agente/Representante:

**DE ELZABURU MÁRQUEZ, Alberto**

**ES 2 568 065 T3**

Aviso: En el plazo de nueve meses a contar desde la fecha de publicación en el Boletín europeo de patentes, de la mención de concesión de la patente europea, cualquier persona podrá oponerse ante la Oficina Europea de Patentes a la patente concedida. La oposición deberá formularse por escrito y estar motivada; sólo se considerará como formulada una vez que se haya realizado el pago de la tasa de oposición (art. 99.1 del Convenio sobre concesión de Patentes Europeas).

## DESCRIPCIÓN

Método y sistema para predicción de enlace en computación móvil

## 5 ANTECEDENTES

La presente invención se refiere a patrones de exploración de enlace en computación móvil. Los enlaces directos futuros son explorados con la ayuda de métodos estadísticos.

10 La computación móvil ha sido propuesta durante mucho tiempo. Ésta incluye la interacción humana y de ordenador. Existen muchos componentes en la computación móvil tal como comunicación, hardware tal como los teléfonos inteligentes, y software tanto en el dispositivo móvil como en el servidor. A pesar del hecho de que el hardware, tal como un teléfono inteligente y el software en los dispositivos móviles han sido desarrollados de forma apropiada, la calidad de la computación móvil es todavía pobre debido a las limitaciones de comunicación de dispositivo-a-dispositivo.

15 En computación móvil, los sistemas que incluyen comunicación de dispositivo-a-dispositivo adoptan diversas formas tal como redes conmutadas de bolsillo (PSN) (Hui, Pan, et al., "Redes conmutadas de bolsillo y movilidad humana en entornos de conferencia", Debates del taller de trabajo 2005 ACM SIGCOMM sobre Redes tolerantes al retardo. ACM, 2005), redes vehiculares, y redes ad-hoc móviles. Tomando PSN como ejemplo, ésta aprovecha los dispositivos móviles como nodos intermedios para retransmitir datos con una cierta tolerancia al retardo. Normalmente no existen rutas directas entre pares de nodos de modo que el enrutamiento en la PSN tiene que llevar a cabo almacenaje, transporte y reenvío, lo que significa que si un nodo desea enviar un mensaje a otro, tiene en primer lugar que almacenar el mensaje y a continuación transportarlo. En cada instantánea de la red, existen algunos nodos que están conectados mientras que otros están desconectados. Por ejemplo, en la Figura 1, los nodos de las zonas sombreadas están conectados entre sí, pero los nodos de áreas diferentes están desconectados. En protocolos de red tradicionales, el envío de paquetes fallará si el nodo de fuente 101 desea enviar un paquete al nodo de destino 104. En PSN, sin embargo, los dispositivos móviles pueden utilizar la movilidad para almacenaje y retransmisión de datos. El paquete procedente de 101 se envía a 102 en el instante t0, y después al nodo 103 en el instante t1. Y por último, el paquete se envía al nodo 104 en el instante t2. Esta aparición y desaparición de enlace dinámico conlleva un montón de oportunidades de comunicación. Pero desafortunadamente, la mayor parte de los protocolos asignan valores estables a métricas de enrutamiento, las cuales harán que la comunicación de extremo a extremo sea ineficaz y nada práctica en un entorno móvil.

35 Realmente resulta difícil diseñar métricas de enrutamiento dinámico para computación móvil debido a la falta de entendimiento de los patrones de enlace. Poca investigación se ha centrado en la ley de aparición y desaparición de enlace. Si se puede predecir la aparición de un enlace futuro, el mecanismo de comunicación entre dispositivos móviles podrá ser optimizado y la métrica de enrutamiento dinámico podrá ser capturada por medio de los protocolos de enrutamiento en caso de que sea posible realizar predicción de enlace en el entorno móvil. Por lo tanto, la predicción en computación móvil es realmente fundamental para garantizar la calidad del servicio y la experiencia de usuario, pero la predicción resulta difícil debido a la falta de un buen entendimiento de la movilidad humana.

45 Algún trabajo previo ha dado algunas sugerencias acerca de las probabilidades de enlace, las cuales se usan como métrica de enrutamiento sin verificación de su racionalidad. El trabajo previo se ha centrado sobre predicción de enlace en redes complejas y en redes sociales. Éstas tratan normalmente con la red modo de gráfico de red estático. Por ejemplo, el documento WO 2003083695 A1 se refiere al problema de predicción de enlace como problema de clasificación binaria y aplica el algoritmo de máquina de vector de soporte para predecir atributos de eventos. Además, Linyuan et al. (Linyuan Lü, Tao Zhou, "Predicción de enlace en redes complejas: Una encuesta", Physica A: Mecánica Estadística y sus Aplicaciones, Volumen 390, Edición 6, 15 de Marzo de 2011, Páginas 1150-1170, ISSN 0378-4371) han listado varias métricas estáticas entre pares y seleccionado las mejor clasificadas para predicción.

55 Además, la computación móvil que incluye comunicación ha demostrado presentar varios atributos específicos humanos en redes complejas o sociales tal como distribución de ley de potencia y vínculos débiles, los cuales hacen que sean aplicables para introducir algunos métodos de predicción de enlace bien realizados en redes complejas y sociales. La predicción de enlace en redes complejas ha sido clasificada en dos partes: algoritmos basados en similitud y algoritmos basados en modelos. La mayor parte de los algoritmos basados en similitud usan algunos índices como vecinos comunes. Éstos son simples de implementar y actúan especialmente bien. Los algoritmos basados en modelos optimizan el rendimiento pero hacen que la computación sea complicada. Todos estos métodos de predicción tratan con la red a modo de instantánea estática de acumulación de historial de enlace, y predicen enlaces desconocidos en vez de la aparición de historiales de enlaces que aparecen y desaparecen. De ese modo, no resulta factible usar directamente esos índices en sistemas de alta movilidad.

65 Ha existido una gran cantidad de demanda de servicios con predicción de enlace en computación móvil. Esas clases de redes son especialmente adecuadas para un gran volumen de transmisión de datos con una alta tolerancia al

retardo. Por ejemplo, compartir contenidos entre dispositivos móviles es una aplicación prospectiva. Si alguien posee una película de pago en su teléfono móvil pero sus amigos también desean visualizarla. El amigo tiene la opción de pagar decenas de dólares por la misma o esperar varias horas para obtenerla. La mayor parte de la gente elegirá lo último puesto que ahorrará dinero. Así, compartir contenidos mediante redes oportunistas es una aplicación prospectiva. También, Han et al. (Han B, Hui P, Kumar V S A et al., Descarga de datos móviles a través de comunicaciones oportunistas y participación social [J]. Computación Móvil, IEEE Transactions on, 2012, 11(5): 821-834) han propuesto un sistema de descarga móvil, el cual despliega redes oportunistas para mitigar el problema de sobrecarga en redes 3G. Para tareas de descarga de tráfico celular, esto puede seleccionar el conjunto objetivo y enviar el tráfico a algún nodo para diseminación de contenidos con red oportunista. Este sistema no puede solamente compartir tráfico pesado de redes celulares, sino que también ahorra un montón de dinero para usuarios móviles. Así, la descarga móvil a través de redes oportunistas es también una aplicación tentadora para el futuro.

El documento US 2009/0310485 A1 se refiere a sistemas y métodos para gestionar de forma efectiva comunicación de red empleando múltiples trayectorias entre sitios.

Además, el consumo de energía de los teléfonos móviles es un factor limitativo en todas esas aplicaciones. Con el fin de suministrar el mensaje a usuarios objetivo, se diseminarán varias copias de mensajes a través de la red, que agotarán la potencia de los dispositivos pronto. Esa mala experiencia de usuario conducirá a los usuarios a rechazar la cooperación en entornos de computación móvil.

Por lo tanto, la presente invención tiene como objetivo predecir de manera directa enlaces en entornos de computación móvil. Es decir, la presente invención está dirigida a predecir enlaces en escenarios como redes vehiculares, redes oportunistas y movilidad humana.

## SUMARIO

La invención está especificada en las reivindicaciones.

Según un aspecto de la presente invención, se proporciona un método y un sistema para predicción de enlace en una red de computación móvil. El método comprende con preferencia romper la red de computación móvil en una secuencia instantáneas en diferentes momentos, representando cada instantánea un patrón de enlace de la red de computación móvil. Además, en base a las instantáneas, el método proporciona datos de histórico de enlace, que representan el historial de nodos conectados en la red de computación móvil. Adicionalmente, en base a los datos de histórico de enlace, el método modela la probabilidad de enlaces en la siguiente instantánea de la red de computación móvil usando regresión kernel. El método comprende además evaluar la probabilidad de enlaces en la siguiente instantánea para obtener una clasificación de probabilidad de enlace, y en base a los resultados el método proporciona a la salida la probabilidad de enlace para cada enlace individual y/o el patrón de enlace pronosticado en la siguiente instantánea.

Con preferencia, el método para predicción de enlace en una red de computación móvil comprende evaluar uno o más índices de clasificación.

Adicionalmente, para evaluar uno o más índices de clasificación el método puede usar al menos uno de entre un índice Academic-Adar y un índice de frecuencia de contacto.

Además, la etapa de provisión de datos de histórico de enlace puede comprender además catalogarlos en registros de contacto directo y en registros de contacto indirecto.

Con preferencia, los registros de contacto directo comprenden un conjunto de datos recopilados a partir de módulos de Bluetooth™ y de Wi-Fi ad hoc en dispositivos móviles. Esos metadatos pueden ser usados directamente como entrada.

Incluso más preferiblemente, los registros de contacto indirecto pueden comprender datos como rutas GPS y archivos de registro de puntos de acceso Wi-Fi. Esos datos no contienen enlaces directos entre pares. De ese modo, necesitan ser transformados en enlaces con determinadas presunciones, por ejemplo que puedan establecer enlaces esos puntos GPS dentro de los 30 m, o los dispositivos móviles registrados en el mismo punto de acceso en el mismo momento.

Además, la etapa de romper la red de computación móvil en una secuencia de instantáneas en diferentes momentos puede comprender además formalizar contactos en cuanto a tiempo de duración en una instantánea de red y una matriz de frecuencia.

Con preferencia, la modelación de la probabilidad de enlaces en la siguiente instantánea de la red de computación móvil usando regresión kernel comprende además definir un conjunto de características para los pares de nodos de la red de computación móvil, modelación de puntos de datos de la red de computación móvil usando regresión kernel, definición de la distancia entre puntos de datos en base al resultado de la regresión kernel, tratamiento con

datos dispersos en los conjuntos de datos obtenidos de acuerdo con las etapas que anteceden, y diseño de un algoritmo de baja complejidad usando un algoritmo de búsqueda de vecino más cercano.

5 Adicionalmente, definir un conjunto de características para los pares de nodos de la red de computación móvil puede comprender además seleccionar al menos dos métricas preferidas para predicción de enlace estático y especificar pares usando esas métricas.

10 Además, la modelación de puntos de datos de la red de computación móvil usando regresión kernel puede comprender además dividir las instantáneas en comunidades con dos catálogos, es decir, detección de comunidad no solapante y detección de comunidad solapante.

Con preferencia, la detección de comunidad no solapante comprende algoritmos que dividen la red de computación móvil en áreas no solapantes, es decir, nodos que pertenecen solamente a una comunidad en cada instantánea.

15 Más preferiblemente, la detección de comunidad solapante comprende algoritmos que dividen la red de computación móvil en áreas solapantes, es decir, los nodos pueden pertenecer a varias comunidades en cada instantánea.

Adicionalmente, dividir las instantáneas en comunidades con dos catálogos puede comprender además transformar las comunidades en matrices computables.

20 Además, definir la distancia entre puntos de datos puede comprender además sumar la distancia de variación total, la cual mide la distancia de dos funciones, entre todos los elementos.

25 Adicionalmente, tratar con datos dispersos puede comprender además agrandar la comunidad local de un nodo y calcular el intervalo normal de confianza para la estimación de probabilidad.

Con preferencia, diseñar el algoritmo de baja complejidad usando un algoritmo de búsqueda de vecino más cercano puede comprender además convertir los puntos modelados en secuencias de bits, y usar hash sensible local para rebajar la complejidad de computación.

30 Por otra parte, se establece un modelo estadístico para predicción de enlace futuro con datos de histórico del enlace. Se define un conjunto de características entre pares de nodos. Se proporcionan dos algoritmos para detectar comunidades de la red. Se identifica una matriz mediante cada nodo para que represente su comunidad local. Se proporciona una definición de la distancia de matriz para medir la distancia entre puntos. Se proporciona un método que trata datos de dispersos. Por otra parte, se introduce un algoritmo de aproximación para rebajar la complejidad con alguna sobrecarga de precisión. Los experimentos demostraron poca pérdida de precisión cuando se desplegó el algoritmo mientras que la complejidad se redujo mucho. El modelo para histórico de enlaces se establece por medio de regresión kernel. La matriz que representa la comunidad local de un nodo se usa como un punto en regresión kernel. Se diseña un algoritmo para búsqueda de vecinos de matriz más cercanos, con una definición de distancia. Por último, se introduce un método para evaluar la efectividad de métricas de enrutamiento.

La presente invención se define mediante las reivindicaciones que siguen, y nada de esta sección deberá ser tomado como limitación respecto a esas reivindicaciones. Los aspectos y ventajas adicionales de la invención se discuten en lo que sigue junto con las realizaciones preferidas.

45 **BREVE DESCRIPCIÓN DE LOS DIBUJOS**

- La Figura 1 ilustra el contexto de predicción de enlace;
- La Figura 2 ilustra un ejemplo de cómo las redes oportunistas envían los datos;
- La Figura 3 ilustra cómo dividir la red en instantáneas;
- 50 La Figura 4 ilustra la descripción detallada sobre el método basado en regresión kernel;
- La Figura 5 ilustra cómo definir el conjunto de características de pares de nodos;
- La Figura 6 ilustra el concepto de regresión kernel;
- La Figura 7 ilustra cómo modelar los puntos para regresión kernel, y
- La Figura 8 ilustra el sistema de predicción de enlace según se ha implementado.

55 **DESCRIPCIÓN DETALLADA DE REALIZACIONES PREFERIDAS**

Ahora se van a describir algunas realizaciones preferidas con referencia a los dibujos. A efectos de explicación, se definen varios detalles específicos, sin apartarse del alcance de la presente invención según se reivindica.

60 Para evitar una mala experiencia de usuario según se ha puesto de relieve con anterioridad, la arquitectura heterogénea que aprovecha predicción de enlace se aplica para mejorar la experiencia de usuario en tales aplicaciones. Para aplicaciones de compartir contenidos, este sistema puede hacer lo siguiente: 1) notificaciones para usuarios acerca de si se pueden satisfacer o no peticiones de contenido, y en qué medida se pueden satisfacer; 2) selección de una trayectoria de enrutamiento para el portador del contenido en caso de que la petición pueda ser satisfecha; 3) promover recomendaciones para usuarios sobre el contenido en el que puedan estar

interesados y clasificar la accesibilidad de esos contenidos. En tareas de descarga de datos móviles, el sistema minimizará los tiempos de transmisión evaluando la conectividad en esas redes. Según la predicción de enlace, este sistema indica tareas de descarga cuyos nodos son la más céntrica de las redes y cuyos nodos no son accesibles por parte de redes oportunistas. Finalmente, el establecimiento de objetivo para tareas de descarga podrá ser seleccionado y los nodos inaccesibles realizarán en cambio transmisión celular.

La presente invención proporciona un método de predicción en computación móvil que tiene, en la realización ilustrada, cinco componentes generales, los cuales han sido mostrados en la Figura 2. Éste incluye componentes de entrada (etapa 201) y de salida (etapa 205), lo que demuestra que el sistema necesita historial de conexión como entrada básica. Tras varias etapas, este método obtiene el resultado de enlaces pronosticados y probabilidad de enlace. Pero debe apreciarse que podrían existir algunos archivos de contacto indirecto tal como registros de GPS y registros de conexión de punto de acceso Wi-Fi en la introducción de datos de historial de enlace. Se deben hacer algunas correcciones en esos datos, por ejemplo asumiendo que los puntos de GPS dentro de los 30 m, o los dispositivos móviles registrados en el mismo punto de acceso al mismo tiempo, pueden establecer enlaces. También se puede apreciar que con el fin de usar el método principal en la etapa 203, los patrones de historial de enlace podrán ser en primer lugar divididos en instantáneas (etapa 202) en diferentes momentos. En definitiva, se hace una evaluación sobre cómo se podría conseguir un mayor perfeccionamiento evaluando algunos índices (etapa 204). En la descripción que sigue, las instantáneas se introducen en primer lugar en la Figura 3. Las etapas de regresión kernel han sido ilustradas en la Figura 4.

Con el fin de hacer frente con dinamicidad, conforme a la etapa 202 de la Figura 2, la red se divide en primer lugar en una secuencia de instantáneas (301, 302, 303, ...) según se muestra en la Figura 3. Cada instantánea representa el patrón de contacto de la red completa en el intervalo de tiempo correspondiente. Podrían existir múltiples bordes entre nodos dado que éstos pueden contactar varias veces durante la duración de una instantánea. Según se muestra también en la Figura 3, las instantáneas pueden ser sustituidas por una matriz de frecuencia que documenta la frecuencia de contacto de pares de nodos. La matriz de frecuencia es realmente una matriz adyacente  $M$  que documenta la cantidad de bordes entre dos pares de nodos (matriz mostrada en la Figura 3). Si el nodo 1 y el nodo 2 conectan dos veces en una instantánea, entonces  $M[1][2]=2$ . Las veces que éstos conectan entre sí se conoce como frecuencia de contacto y la matriz correspondiente de todos los pares de nodos se conoce como matriz de frecuencia.

La etapa 203 de la Figura 2 requiere modelar el sistema usando regresión kernel. El método basado en regresión kernel contiene cinco partes en la Figura 4, que incluyen definir un conjunto de características (etapa 401), modelar el punto (etapa 402), definir la distancia según se muestra en la etapa 403, tratar datos dispersos (etapa 404) y diseñar un algoritmo de aproximación (etapa 405). Las etapas van a ser descritas con mayor detalle en lo que sigue.

#### DEFINIR EL CONJUNTO DE CARACTERÍSTICAS (etapa 401)

En cada instantánea existe información estructural, la cual es suficiente para la predicción de enlaces en computación móvil. Por ejemplo, en la Figura 5 (una instantánea obtenida en la etapa 202), el nodo 501 y el nodo 502 pueden constituir atributos como vecinos comunes, la mínima cantidad de salto, y su frecuencia de contacto. En este caso, los vecinos comunes entre sí son 2 (503 y 504). La mínima cantidad de salto es 1, lo que significa que no existen nodos de retransmisión sobre la trayectoria más corta entre los nodos 501 y 502. La frecuencia de contacto es 3, lo que significa que éstos contactan tres veces durante el período de tiempo de la instantánea. Como ejemplo adicional, los nodos 505 y 503 forman los siguientes atributos: 0 vecinos comunes, la cantidad de salto mínimo es 1, y la frecuencia de contacto es 2. En decisiones de predicción gráfica estática, se usan estas métricas como clasificación para que consideren que, cuanto más altos son esos índices, más alta es la probabilidad de que puedan establecer enlaces. De ese modo, se seleccionarían como enlaces pares clasificados en posición más alta, los que aparecerán más probablemente a continuación.

En el ámbito de predicción kernel, sin embargo, resulta redundante usar demasiada información estructural en gráficos estáticos. Así, el sistema elige preferiblemente algunas métricas con mejor rendimiento. Conforme a algunos experimentos sobre instantáneas de conjuntos de datos móviles, se ha encontrado que Academic-Adar (AA) y frecuencia de contacto se comportan especialmente bien. AA es un método basado en similitud para predicción de enlace estático. La idea básica para ello es contar los vecinos comunes entre dos nodos con algún peso. Lo que contribuye un vecino común al índice de clasificación para un par, se determina mediante el grado de este vecino común. Cuanto más alto sea el grado, menos contribuye este vecino común al índice de clasificación final. Se puede encontrar una descripción detallada del índice de AA en Linyuan Lü, Tau Zhou, Predicción de enlace en redes complejas: Una encuesta, Physica A: Mecánica Estadística y sus Aplicaciones, Volumen 390, edición 6, 15 de Marzo de 2011, Páginas 1150-1170, ISSN 0378-4371. La frecuencia de contacto puede mostrar directamente la posibilidad de enlaces, dado que si un par contacta de forma muy frecuente en la instantánea actual, es muy probable que éstos comuniquen entre sí en el futuro. Estos dos índices se calculan con poca complejidad y pueden alcanzar una alta precisión en instantáneas estáticas. De ese modo, resulta razonable usarlos como posible base de referencia para modelar el sistema de predicción con regresión kernel.

Puesto que han sido seleccionados los dos mejores índices de clasificación, se puede hacer una definición para el

conjunto de características de dos nodos ( $i$  y  $j$ ) en el intervalo de tiempo  $t$ .  $f_t(i, j) = \{aa_t(i, j), cf_t(i, j)\}$ , que se selecciona a partir de un conjunto finito  $F$  de características. En este caso,  $aa_t(i, j)$  y  $cf_t(i, j)$  representan respectivamente el valor AA y la frecuencia de contacto entre  $i$  y  $j$  en la instantánea  $G_t$ . Según la suposición de que las características de la estructura de red determinan la probabilidad de enlace, dos pares en cualquier instantánea tendrán una probabilidad similar de enlace si éstos comparten el mismo conjunto de características  $f$ . La única diferencia entre ambos se determina mediante su estructura de red local. Se define  $B_t(f)$  como el número de pares de nodos que tienen las mismas características  $f$  desde el momento de inicio hasta el momento actual  $t$ , y  $B_t^+(f)$  como el número de tales pares que tienen enlaces en la siguiente instantánea. En base a estimación de punto, la probabilidad de enlace futuro de pares en la instantánea actual  $G_t$  deberá ser:

$$P_t(i, j) = p(f_t(i, j)) = \beta_t(f) / \beta_t^+(f). \quad (1)$$

Si la probabilidad de enlace se estima simplemente de esta manera, la probabilidad podría ser muy desigual e imprecisa. En particular, muchos de los puntos de datos estadísticos pueden producir gran cantidad de ruido que finalmente perturbará la probabilidad de enlace real. Por ejemplo, algunos pares que comparten las mismas características tienen estructura de red local completamente diferente, mientras que otros tienen una estructura muy similar, pero contribuyen lo mismo en la ecuación 1. Así, debería existir algún método estadístico para corregir este defecto, el cual se aplica en lo que sigue.

#### 20 MODELAR PUNTOS DE DATOS EN REGRESIÓN KERNEL (etapa 402)

La regresión kernel es un método estadístico no paramétrico a efectos de estimación. El objetivo es encontrar la relación no lineal entre dos variables. La Figura 6 ilustra cómo se usa la regresión kernel para estimar el punto desconocido 601. Está claro que la línea 605 no es lineal y no puede ser representada mediante ninguna función, lo que hace que la estimación para el valor  $y$  del punto 601 sea imposible. Pero en regresión kernel, se pueden aprovechar los puntos conocidos en torno al punto estimado para especular con el desconocido. Existen muchos puntos como el punto 602 y el punto 603 en torno al punto desconocido 601. Éstos tienen un valor  $y$  muy similar al del punto 601, y por tanto pueden ser usados para estimar el punto 601.

Sin embargo, dado que los puntos tienen distancias diferentes al punto estimado, éstos podrían tener diferentes contribuciones a la estimación. Por ejemplo, el punto 602 está próximo al punto 601, pero el punto 603 está lejos del mismo. Por lo tanto, se necesita poner más peso sobre el valor  $y$  del punto 602 que del punto 603. Finalmente, tras tomar en consideración todos los vecinos del interior del círculo 604, se obtiene el resultado estimado para el valor del punto 601. El punto principal de esta estimación consiste en evaluar la proximidad entre puntos de datos. En computación móvil, la predicción de enlace falla realmente en cuanto a predecir la posibilidad de que aparezca el enlace. Por lo tanto, el valor estimado deberá ser la probabilidad de enlace. La proximidad entre enlaces, sin embargo, es difícil de medir.

En estadísticas no paramétricas, existen muchos estimadores para regresión kernel. Realmente, la mayor parte de ellos, como los estimadores kernel de Priestly-Chao, Grassler-Müller y Nadaraya-Watson, pueden proporcionar buenas estimaciones. El sistema de la invención toma como ejemplo el estimador kernel de Nadaraya-Watson para predecir enlaces. Lo más importante en la estimación kernel de Nadaraya-Watson es encontrar la función kernel óptima, lo que puede hacer que el valor estimado se aproxime al valor real. La función kernel de Nadaraya-Watson se introduce aquí para atribuir menos peso a puntos de datos ruidosos y más peso a puntos de datos muy cercanos. El estimador de Nadaraya-Watson se lee como:

$$\hat{m}_h(x) = \frac{\sum_{i=1}^n K_h(x - X_i) Y_i}{\sum_{i=1}^n K_h(x - X_i)}, \quad (2)$$

donde  $\hat{m}_h(x)$  es el valor estimado de  $y$  cuando  $X=x$ .  $K_h(x-X_i)$  es la función kernel con un ancho de banda  $h$ . Esta función define la proximidad entre el punto de datos actual ( $x, y$ ) y el punto histórico ( $X_i, Y_i$ ), y el ancho de banda  $h$  determina lo suave que será el proceso de estimación. En este contexto de predicción de enlace, la probabilidad estimada se define como:

$$\hat{p}_t(i, j) = \frac{\sum_{x, y, t'} S(f_t(i, j), f_{t'}(x, y)) \cdot L_{t'+1}(x, y)}{\sum_{x, y, t'} S(f_t(i, j), f_{t'}(x, y))}, \quad (3)$$

donde  $L_{t'+1}(x, y) = 1$ , e  $y$  tienen enlaces en el instante  $t'+1$ , en otro caso,  $L_{t'+1}(x, y) = 0$ . Pero un reto consiste en evaluar la proximidad de cada uno de los pares ( $x$  e  $y$ ) al par actual ( $i$  y  $j$ ) cuando se toma en consideración la complejidad de búsqueda. Por lo tanto, kernel  $S^*$  se redefine para simplificar la ecuación 3. Designese  $K(cf_t(i, j))$ ,

$c_t'(x, y) \cdot I_{f_t(i,j)}(f_t(x, y))$  como  $S^*$ . En este caso,  $K(c_t(i, j), c_t'(x, y))$  es la nueva función kernel y  $c_t(i, j)$  va a ser explicada más adelante. Así, la  $p_t(i, j)$  puede ser reescrita como:

$$\begin{aligned} & \frac{\sum_{x,y,t'} K(c_t(i, j), c_t'(x, y)) \cdot I_{f_t(i,j)}(f_{t'}(x, y)) \cdot L_{t'+1}(x, y)}{\sum_{x,y,t'} K(c_t(i, j), c_t'(x, y)) \cdot I_{f_t(i,j)}(f_{t'}(x, y))} \\ &= \frac{\sum_{x,y,t'} K(c_t(i), c_t'(x)) \cdot I_{f_t(i,j)}(f_{t'}(x, y)) \cdot L_{t'+1}(x, y)}{\sum_{x,y,t'} K(c_t(i), c_t'(x)) \cdot I_{f_t(i,j)}(f_{t'}(x, y))} \\ &= \frac{\sum_{x,t'} K(c_t(i), c_t'(x)) \cdot \sum_y I_{f_t(i,j)}(f_{t'}(x, y)) \cdot L_{t'+1}(x, y)}{\sum_{x,t'} K(c_t(i), c_t'(x)) \cdot \sum_y I_{f_t(i,j)}(f_{t'}(x, y))} \\ &= \frac{\sum_{x,t'} K(c_t(i), c_t'(x)) \cdot \beta_{it'}^+(f_t(i, j))}{\sum_{x,t'} K(c_t(i), c_t'(x)) \cdot \beta_{it'}(f_t(i, j))}, \quad (4) \end{aligned}$$

$$K(c_t(i), c_t'(x)) = \exp\left(-\frac{c_t(i) - c_t'(x)}{2h^2}\right). \quad (5)$$

5 En este caso,  $K(c_t(i, j), c_t'(x, y))$  es la kernel Gaussiana en la ecuación 4, y  $c_t(i)$  es la comunidad del nodo  $i$  en el instante  $t$ .  $\beta_{it}(f_t(x, y))$  es el número de pares en la comunidad de  $i$  en la instantánea  $t$  con la característica  $f_t(x, y)$  mientras que  $\beta_{it}^+(f_t(x, y))$  es el número de tales pares que tienen un enlace en la siguiente instantánea.

10 En la kernel Gaussiana, existen dos parámetros importantes: el ancho de banda  $h$  y la distancia entre comunidades  $c_t(i)$  y  $c_t'(i)(|c_t(i) - c_t'(x)|)$ . En regresión kernel, la elección del ancho de banda es vital puesto que si  $h$  es demasiado grande, el ajuste será muy suave, y si  $h$  es demasiado pequeña, el ajuste será muy basto. El ancho de banda óptimo está comprendido en una gama determinada y se puede conseguir el ancho de banda óptimo mediante puntuación

15 de validación cruzada  $cv = \hat{R}(h) = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{r}_{-i}(x_i))^2$  donde  $\hat{r}_{-i}(x_i)$  es el estimador obtenido omitiendo el punto  $i$ -ésimo (Wasserman, Larry. Todas las estadísticas no paramétricas. Vol. 4, núm. 6, Nueva York; Springer, 2006).

De hecho, existen muchos algoritmos de detección de comunidad, todos los cuales pueden ser aplicados a este sistema. Estos algoritmos pueden dividirse en dos categorías: detección de comunidad no solapante, por ejemplo CNM (Clauset, Aaron Mark EJ Newman, y Christopher Moore. "Encontrar estructura de comunidad en redes muy grandes". Revista Physical E 70.6 (2004): 066111), y detección de comunidad solapante, por ejemplo grupo-k (Palla, Gergely, et al., "Descubrir la estructura de comunidad de solapamiento de redes complejas en la naturaleza y la sociedad". Nature 435.7043 (2005): 814-818). En una red con un número de nodos muy grande, es razonable usar la detección de comunidad no solapante, debido a que es muy eficaz en el tiempo detectar comunidades y cuantificar comunidades como métricas para medir la distancia de comunidad. Si solamente existe un pequeño número de nodos, el sistema deberá aplicar el método de detección de comunidad solapante para superar el problema de datos dispersos descrito en la etapa 404. Además, si existe una demanda estricta por complejidad de tiempo, se deberá aplicar el método de detección de comunidad en tiempo real (Leung, Ian XY, et al. "Hacia la detección de comunidad en tiempo real en grandes redes". Revista Physical E 79.6 (2009): 066107). En este caso, se mencionan dos algoritmos como ejemplos: CNM, un algoritmo no solapante, y grupo-k, un algoritmo solapante.

30 En algoritmo CNM, se supone que una comunidad en una instantánea deberá tener más enlaces entre sí que con cualesquiera otros nodos del gráfico. Una buena medida para cuantificar todo esto es la modularidad. La definición de modularidad es:

$$Q = \frac{1}{2m} \sum_{ij} \left[ l_{ij} - \frac{k_i k_j}{2m} \right] \delta(d_i, d_j), \quad (6)$$

35 donde  $m$  es el número de enlaces en la instantánea,  $l_{ij} = 1$  si existe un enlace entre ellos, en otro caso  $l_{ij} = 0$  y  $\delta(d_i, d_j) = 1$  si  $d_i = d_j$ , lo que significa que  $i$  y  $j$  pertenecen a la misma comunidad; en otro caso,  $\delta(d_i, d_j) = 0$ . Esta modularidad  $Q$  indica la racionalidad de la partición de comunidad: cuanto más alta sea  $Q$ , mejor se detectan las comunidades. La idea principal de esta algoritmo es encontrar la máxima modularidad global. Pero encontrar la modularidad óptima es un problema arduo de NP. A diferencia del algoritmo descrito en el documento WO

2003083695 A1, este método de predicción hace uso de un algoritmo de anillamiento simulado para aproximar la modularidad óptima. Inicialmente todos los nodos pertenecen a su propia comunidad y los nodos se hacen converger a la misma comunidad con una distribución exponencial. La modularidad óptima cercana se adquirirá después de varias iteraciones. La descripción detallada puede ser apreciada en el Algoritmo 1:

```

5
1 Inicializar  $N$  comunidades, inicializar el número requerido de muestras  $numS$ 
2 Asignar la totalidad de los  $N$  nodos a una de las  $N$  comunidades aleatoriamente
3 obtener  $Q_i$  conforme a la ecuación 6
4  $i=1$ ;
10 5 mientras  $i < numS$ ,
6     seleccionar un nodo  $n$  aleatoriamente;
7     mover  $n$  desde su comunidad actual hasta otra comunidad elegida aleatoriamente
8     calcular una nueva  $Q_i$  y  $\Delta Q_i = Q_i - Q_{i-1}$ 
9     si  $\Delta Q_i \leq 0$  entonces
15 10         aceptar el movimiento del nodo con una probabilidad de  $exp(\Delta Q)$ .
11         si se acepta el movimiento, entonces
12              $i^{++}$ 
13         endif
14     endif
20 15 en otro caso
16     aceptar el movimiento del nodo
17          $i^{++}$ 
18 endelse
19 endwhile
25 20 presentar a la salida la comunidad final detectada
21 retorno
    
```

En el algoritmo de grupo-k se define, no obstante, una buena comunidad como múltiples sub-gráficos que tienen muchos vecinos comunes. Y los grupos del grupo-k se funden en una agrupación. De ese modo, en esta clase de partición de comunidad, un solo nodo puede tener la oportunidad de pertenecer a varias comunidades, y por lo tanto éste es un método de detección de comunidad solapante.

DEFINICIÓN DE DISTANCIA DE COMUNIDAD (etapa 403)

En las comunidades, es difícil definir las distancias entre ellas, debido a que no están identificadas numéricamente. Al igual que en lo descrito con anterioridad, los pares en las comunidades son también especificados con conjuntos de características estructurales, es decir en las comunidades existen solamente conjuntos de características sin números de nodo. De ese modo, es posible transformar la comunidad  $c_t(i)$  en una representación de matriz de dos

dimensiones:  $c_t(i) = \{\beta_{it}^+(f), \beta_{it'}^+(f) \forall f \in F\}$  (Figura 7). En este caso, esta matriz de dos dimensiones está indexada por el conjunto finito de características  $F$ . Cada elementos de esta matriz contiene dos valores:  $\beta_{it}^+(f)$  y  $\beta_{it'}^+(f)$ .  $\beta_{it}^+(f)$  es el número de pares en la comunidad de  $i$  en la instantánea  $t$  (con la característica  $f_i(x, y)$  mientras que  $\beta_{it'}^+(f)$  es el número de tales pares que tienen un enlace en la siguiente instantánea. Cada elemento en  $c_t(i)$  puede calcular la probabilidad de enlace  $p(f)$  bajo la característica  $f$ . La matriz es realmente un espacio de probabilidad de dos dimensiones. Así, la diferencia entre matrices puede ser la totalidad de la diferencia de probabilidad de cada elemento. Pero dicha estimación de punto no representa la variación en la probabilidad también. Por lo tanto, se usa en cambio la distribución posterior  $p(f)$  y ésta es una distribución beta con la expectativa de  $\beta^+(f) / \beta(f)$ , puesto que el aspecto de un enlace es una distribución binomial con parámetro  $p(f)$ . Así, cada elemento de las matrices es ahora una distribución beta, y la proximidad de dos matrices se define como:

$$|c_t(i) - c_{t'}(x)| = \sum_{f \in F} \delta(\pi_1, \pi_2) \quad (7)$$

$$\pi_1 \sim B(\beta_{it}^+(f), \beta_{it} - \beta_{it}^+(f))$$

$$\pi_2 \sim B(\beta_{xt'}^+(f), \beta_{xt'} - \beta_{xt'}^+(f)),$$

50 donde  $\delta(\pi_1, \pi_2)$  es la distancia de variación total que mide la distancia de dos funciones. Combinando (4)(5)(7) se puede calcular la probabilidad de enlace  $\hat{p}_t(i,j)$ .

TRATAR CONJUNTOS DE DATOS DISPERSOS (etapa 404)

Los datos de enlace son muy dispersos en muchos conjuntos de datos, lo que hace que la matriz de comunidad sea

muy dispersa con muchos elementos vacíos. Por lo tanto, la comunidad de un nodo particular debería ser extendida con el componente conectado. Ahora, la comunidad local de un nodo es la comunidad real detectada a partir del algoritmo CNM y del gráfico conectado al que pertenece. Aparte de esto, el conjunto de características deberá ser también contado de nuevo a efectos de aproximación. Cuando se ha construido la matriz, el elemento no sólo deberá ser el conteo exacto de la característica  $f$  correspondiente, sino también las características del interior de un círculo de diámetro  $d$  alrededor del mismo. Por ejemplo, el valor original en  $c_i(i)[2][2]$  (701 en la Figura 7), deberá ser el número de pares de nodos con el conjunto de características  $\langle 2, 2 \rangle$ . Pero si  $d$  está establecido en 1, los elementos del conjunto de características  $c_i(i)[1][2]$ ,  $c_i(i)[2][3]$  y  $c_i(i)[3][2]$  se añaden también a  $c_i(i)[2][2]$ . De ese modo, tras el ajuste, el elemento deberá ser  $2/5$ .

También, en los datos estadísticos usados en la presente memoria, solamente existe un pequeño número de puntos, lo que da como resultado una desviación del valor estimado y algo de inestabilidad. Para resolver este problema, la probabilidad final deberá introducir un intervalo de confianza en vez de una determinada probabilidad. Con preferencia, se introduce un intervalo de confianza normal del 95% para proporcionar una gama de probabilidad para los resultados finales.

#### ALGORITMO DE BÚSQUEDA DE VECINO MÁS CERCANO (etapa 405)

Actualmente, existe un gran número de puntos que deben ser considerados por el método de estimación, y por lo tanto el consumo de tiempo resulta ser un problema. Para rebajar la complejidad de computación en este sistema, se necesita encontrar los vecinos más cercanos de determinadas matrices. Si se pueden encontrar algunos de los puntos de vecino cercano de los estimados, la complejidad de tiempo se podría reducir a una complejidad de tiempo constante. La función hash local sensible es tal como un algoritmo para dispersar los elementos similares en la misma tabla. Para función hash local sensible de secuencias de bits, los elementos con una distancia de Hamming pequeña serán dispersados en la misma tabla con alta probabilidad. Es decir, a pesar del hecho de que el algoritmo no puede dar resultados precisos, éste puede presentar a la salida los vecinos más cercanos de una manera muy eficiente. La complejidad de tiempo para la construcción de la tabla es lineal mientras que el retorno de los vecinos más cercanos es constante en el tiempo.

En el modelo de regresión kernel, con el fin de obtener las matrices más cercanas, es necesario transformar la matriz en secuencias de bits. Cada elemento de la matriz está representado por una función beta continua, la cual puede ser dividida en  $n$  intervalos iguales. Cada masa de probabilidad  $m$  de cada intervalo se obtiene por integración. Ahora se puede representar la función beta continua como una secuencia discreta  $\langle m_1, m_2, m_3, \dots, m_n \rangle$ . Entonces resulta fácil cambiar esta secuencia a una secuencia de bits puesto que cada probabilidad puede estar representada por  $q$  bits, donde los primeros  $[mq]$  bits se establecen en 1, mientras que los restantes se establecen en 0. Finalmente, la matriz en su totalidad puede ser establecida como una secuencia de bits colocando cada elemento ordenadamente.

El algoritmo 2 presenta un algoritmo hash sensible local (LSH) de ese tipo:

```

1  Introducir  $n+1$  secuencias de bits  $\langle B_1, B_2, B_3, \dots, B_n, B_{n+1} \rangle$ , un conjunto vacío S. En la salida el conjunto
S de vecinos más cercanos de  $B_{n+1}$ 
2  int hasNum=35, count=0;
3  mientras count<hasNum
4      seleccionar k bits aleatorios como función hash  $H_{count}$ 
5      count++;
6  endwhile
7  count=0, m=0
8  while m<n
9      para count<hasNum;count++
10         almacenar  $B_m$  en la tabla de dispersión con clave hash  $H_{count}(B_m)$ 
11     endfor
12     count++;
13 endwhile
14 count=0
15 mientras count<hasNum
16     almacenar  $B_{n+1}$  en la tabla hash con clave hash  $H_{count}(B_m)$ 
17     añadir secuencias con la misma clave hash  $H_{count}(B_m)$  en S
18 endwhile
19 retorno

```

En este algoritmo existirán 35 tablas hash diferentes. Cada punto encontrará su posición en estas tablas. Las funciones hash son fáciles de encontrar debido a que este algoritmo selecciona varios bits aleatorios como clave hash. Así, si el punto tiene la misma clave hash que un punto existente en la misma posición, estos podrían ser vecinos muy cercanos.

Hasta ahora, se ha logrado una probabilidad de enlace estimado con baja complejidad, pero no ha existido ninguna evaluación sobre su precisión (etapa 204). En otras palabras, el método necesita clasificar las probabilidades en orden descendente. En primer lugar, se eligen las  $n$  probabilidades más altas (siendo  $n$  dependiente de cuántos de los enlaces se desea predecir en el siguiente intervalo de tiempo). Esos  $n$  pares correspondientes se consideran como los enlaces en el siguiente intervalo de tiempo. Esos enlaces forman el patrón de enlace pronosticado en el siguiente intervalo de tiempo.

Existen varias métricas para evaluación de precisión y se han introducido dos métricas como ejemplo: una intuitiva y una acumulativa. Precisión:  $Precisión = m/n$ , que considera la relación de bordes correctos ( $m$ ) en el número de bordes ( $n$ ) elegido. Por conveniencia,  $n$  se elige de modo que sea el número de enlaces en el siguiente intervalo de tiempo. Este índice evalúa los enlaces clasificados en posiciones más altas, lo que es muy significativo para la predicción de enlace directo.

Área bajo la curva (AUC): El problema de predicción de enlace puede ser considerado como un problema de clasificación binaria en aprendizaje automático. La puntuación de AUC es equivalente a la probabilidad de clasificación correcta. Por ejemplo, si un par con enlace y un par sin enlace se eligen aleatoriamente, la puntuación de AUC es equivalente a la probabilidad de que estos dos pares sean clasificados correctamente. Así, este índice toma en consideración todos los pares de la clasificación y los evalúa como un conjunto.

## IMPLEMENTACIÓN DEL SISTEMA

Según se ha mostrado en la Figura 8, el sistema de predicción de enlace se implementa con arquitectura heterogénea: capa 801 de dispositivo-a-dispositivo, capa 802 de redes celulares y capa 803 de servidor. La capa de dispositivo-a-dispositivo es responsable del gran volumen de transmisión de datos mientras que las redes celulares se usan para transmisión de información valiosa. En la capa de servidor, se realiza control y predicción centralizados.

En la capa de dispositivo-a-dispositivo, los usuarios con redes de teléfono móvil forman redes ad hoc. Existen dos tipos de diseño físico para comunicación: Wi-Fi y Bluetooth. La elección del diseño físico depende de la demanda de transmisiones. Normalmente, la cobertura de Wi-Fi es de aproximadamente 100 metros, mientras que con Bluetooth solamente se puede alcanzar 10 metros. Los usuarios de teléfonos móviles son con frecuencia reacios a abrir su módulo de Bluetooth mientras mantienen el módulo Wi-Fi encendido. Esas propiedades hacen que las conexiones de Wi-Fi sean más densas que las de Bluetooth. Pero el consumo de energía de redes Wi-Fi es enorme comparado con las de Bluetooth. Así, si los archivos transmitidos a través de redes oportunistas son comparativamente pequeños y no tan urgentes, se recomienda seleccionar Bluetooth como dispositivo de transmisión a efectos de ahorrar energía. Pero en otros casos, se recomienda usar Wi-Fi para la transmisión.

La capa de redes celulares es el puente entre el servidor centralizado y los usuarios de teléfonos móviles. En general, existen dos tipos de información transmitida a través de redes celulares: 1) los registros de conexión que incluyen Wi-Fi y Bluetooth se transmiten desde teléfonos móviles hasta el servidor. Los acuses de recibo de los paquetes se envían al servidor; 2) los servidores impulsan recomendaciones a los usuarios de teléfonos móviles a través de redes celulares con anterioridad a que todos los mensajes sean transmitidos.

En cuanto a la capa de servidor, ésta pronosticará patrones de enlace futuro conforme a todos los registros de historial de conexión. Estos patrones de enlace pronosticados ayudarán al servidor a evaluar la calidad futura del servicio. También los acuses de recibo extraídos de los receptores de paquetes serán enviados al servidor, lo que podría ayudar a la transmisión completa del contenido de servicio.

Otros aspectos, características y ventajas resultarán evidentes a partir del sumario que antecede, así como a partir de la descripción que sigue, incluyendo las figuras y las reivindicaciones.

Mientras que la invención ha sido ilustrada y descrita en detalle en los dibujos y en la descripción que antecede, tales ilustración y descripción deben ser consideradas como ilustrativas o ejemplares, y no como limitativas. Se comprenderá que los expertos en la materia pueden realizar cambios y modificaciones dentro del alcance de las reivindicaciones siguientes. En particular, la presente invención cubre otras realizaciones con cualquier combinación de características a partir de las diferentes realizaciones descritas con anterioridad y en lo que sigue.

Además, en las reivindicaciones, la palabra "que comprende" no excluye a otros elementos o etapas, y el artículo indefinido "un" o "una" no excluye una pluralidad. Una sola unidad puede cumplir las funciones de varias características mencionadas en las reivindicaciones. Los términos "esencialmente", "alrededor de", "aproximadamente" y similares en relación con un atributo o un valor, definen también en particular el atributo exactamente o exactamente el valor, respectivamente. Cualquier signo de referencia en las reivindicaciones no debe ser entendido como limitación del alcance.

**REIVINDICACIONES**

- 1.- Un método para predicción de enlace en una red de computación móvil, que comprende:
- 5 a) romper la red de computación móvil en una secuencia de instantáneas (301, 302, 303) en diferentes momentos, representando cada instantánea (301, 302, 303) un patrón de enlace de la red de computación móvil;
- 10 b) en base a las instantáneas (301, 302, 303) obtenidas en la etapa a), proporcionar datos de histórico de enlace que representan el histórico de nodos conectados (501, 502, 503, 504, 505) en la red de computación móvil;
- c) en base a los datos de histórico de enlace obtenidos en la etapa b), modelar la probabilidad de enlaces en la siguiente instantánea de la red de computación móvil usando regresión kernel;
- d) clasificar la probabilidad de enlaces obtenida conforme a la etapa c) en un orden descendente para obtener un patrón de enlace en siguiente instantánea, y
- 15 e) en base a los resultados obtenidos en las etapas c) y d), presentar a la salida la probabilidad de enlace para cada enlace individual y/o el patrón de enlace pronosticado en la siguiente instantánea.
- 2.- El método de la reivindicación 1, en donde el método para predicción de enlace en una red de computación móvil comprende evaluar uno o más índices de clasificación.
- 20 3.- El método de la reivindicación 2, en donde evaluar uno o más índices de clasificación utiliza al menos uno de entre un índice de Academic-Adar y un índice de frecuencia de contacto.
- 4.- El método de cualquiera de las reivindicaciones 1 a 3, en donde la etapa de proporcionar datos de histórico de enlace comprende catalogarlos en registros de contacto directo, los cuales pueden ser usados directamente como entrada, y en registros de contacto indirecto, los cuales no contienen enlaces directos entre pares de nodos.
- 25 5.- El método de la reivindicación 4, en donde los registros de contacto directo comprenden un conjunto de datos recopilados a partir de módulos de Bluetooth y de Wi-Fi ad hoc en los dispositivos móviles, en donde estos metadatos pueden ser preferiblemente usados directamente como entrada.
- 30 6.- El método de la reivindicación 4 ó 5, en donde los registros de contacto indirecto comprenden datos como rutas GPS y archivos de registro de puntos de acceso Wi-Fi.
- 35 7.- El método de cualquiera de las reivindicaciones 1 a 6, en donde romper la red de computación móvil en una secuencia de instantáneas (301, 302, 303) en diferentes momentos comprende formalizar contactos con duración de tiempo en una instantánea de red y una matriz de frecuencia.
- 40 8.- El método de cualquiera de las reivindicaciones 1 a 7, en donde modelar la probabilidad de enlaces en la siguiente instantánea de la red de computación móvil usando regresión kernel, comprende:
- c1) definir un conjunto de características para los pares de nodos de la red de computación móvil;
- c2) modelar puntos de datos de la red de computación móvil usando regresión kernel;
- 45 c3) definir la distancia entre puntos de datos conforme a la etapa b);
- c4) tratar datos dispersos en los conjuntos de datos obtenidos, conforme a las etapas c1) – c3), y
- c5) diseñar un algoritmo de baja complejidad usando un algoritmo de búsqueda de vecino más cercano.
- 9.- El método de la reivindicación 8, en donde definir un conjunto de características para los pares de nodos de la red de computación móvil comprende seleccionar al menos dos métricas preferidas para predicción de enlace estático y especificar pares usando esas métricas.
- 50 10.- El método de las reivindicaciones 8 a 9, en donde modelar puntos de datos de la red de computación móvil usando regresión kernel comprende dividir las instantáneas (301, 302, 303) en comunidades con dos catálogos: detección de comunidad no solapante, y detección de comunidad solapante.
- 55 11.- El método de la reivindicación 10, en donde la detección de comunidad no solapante comprende algoritmos que dividen la red de computación móvil en áreas no solapantes.
- 60 12.- El método de las reivindicaciones 10 a 11, en donde la detección de comunidad solapante comprende algoritmos que dividen la red de computación móvil en áreas solapantes.
- 13.- El método de cualquiera de las reivindicaciones 10 a 12, en donde dividir las instantáneas (301, 302, 303) en comunidades con dos catálogos comprende además transformar las comunidades en matrices computables.
- 65 14.- El método de cualquiera de las reivindicaciones 8 a 13, en donde definir la distancia entre puntos de datos

comprende sumar la distancia de variación total, la cual mide la distancia de dos funciones, entre todos los elementos.

5 15.- El método de cualquiera de las reivindicaciones 8 a 14, en donde tratar datos dispersos comprende agrandar la comunidad local de un nodo y calcular el intervalo de confianza normal para la estimación de probabilidad.

10 16.- El método de cualquiera de las reivindicaciones 8 a 15, en donde diseñar un algoritmo de baja complejidad usando un algoritmo de búsqueda de vecino más cercano comprende convertir los puntos modelados en secuencias de bits, y usar hash sensible local para rebajar la complejidad de computación.

17.- Un sistema para predicción de enlace en computación móvil configurado para llevar a cabo el método de cualquiera de las reivindicaciones 1 a 16.

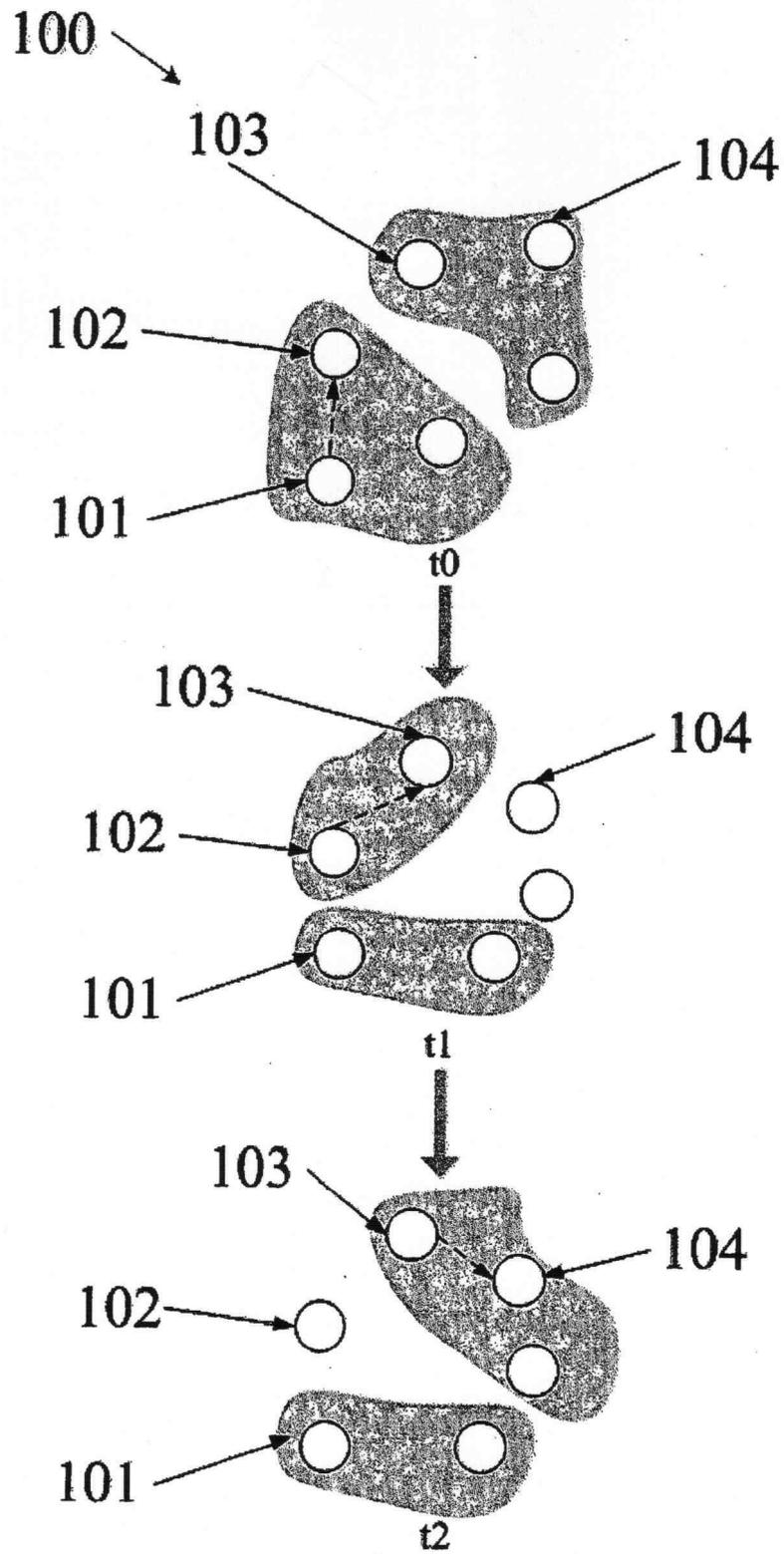


FIG. 1

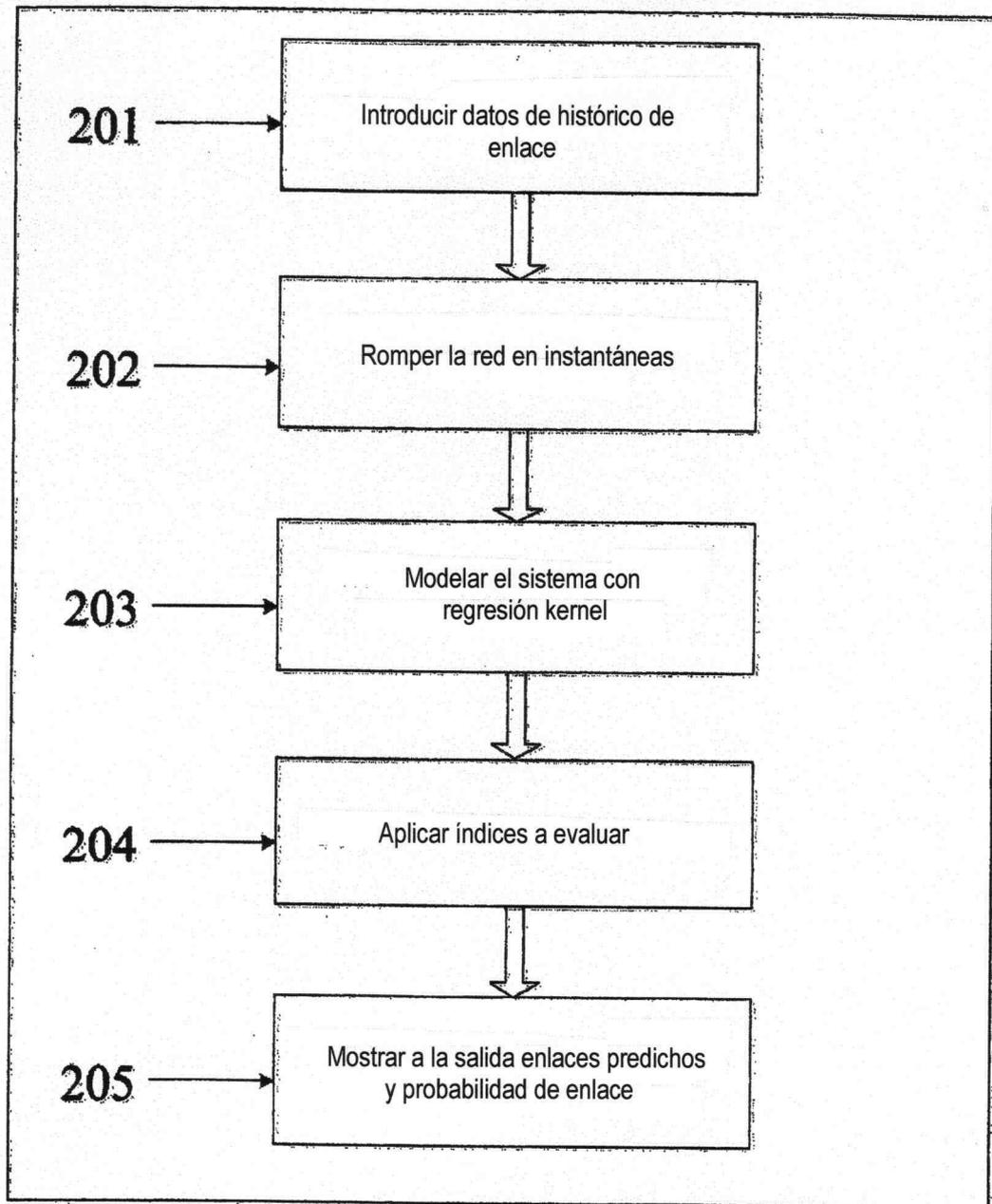


FIG. 2

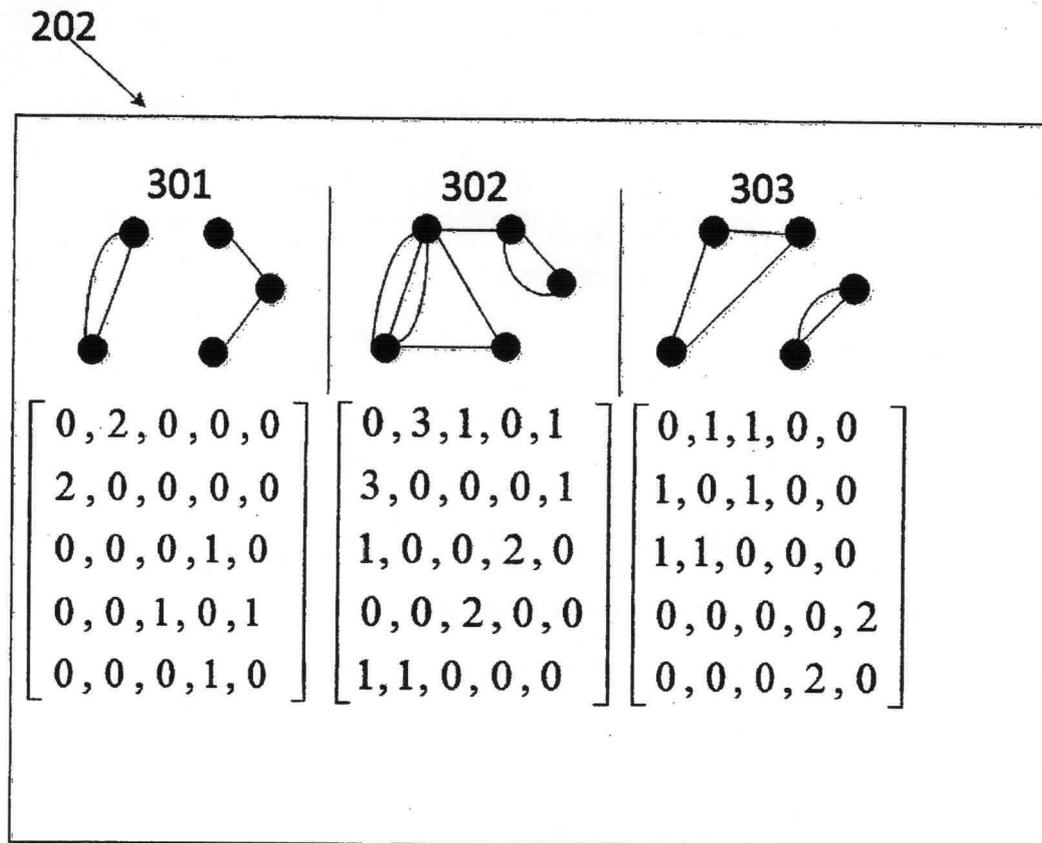


FIG. 3

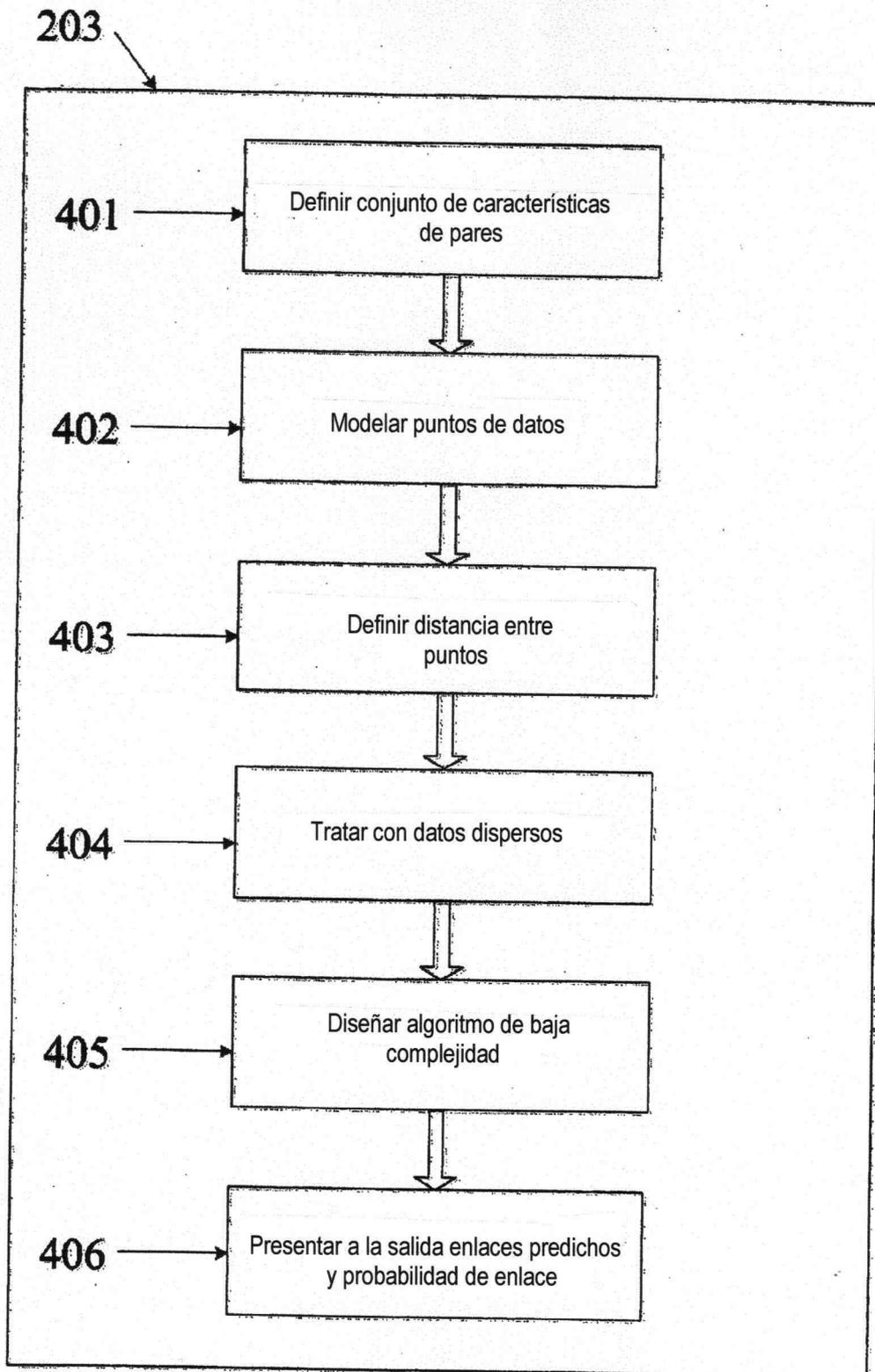


FIG. 4

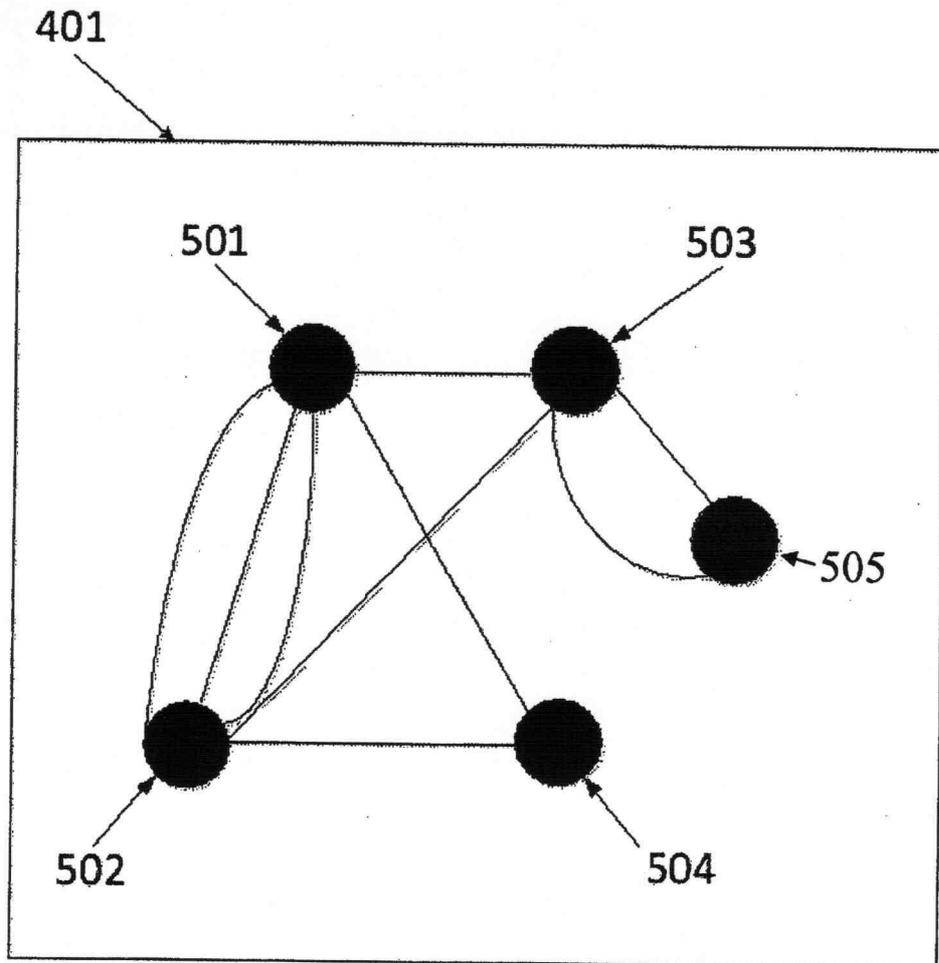


FIG. 5

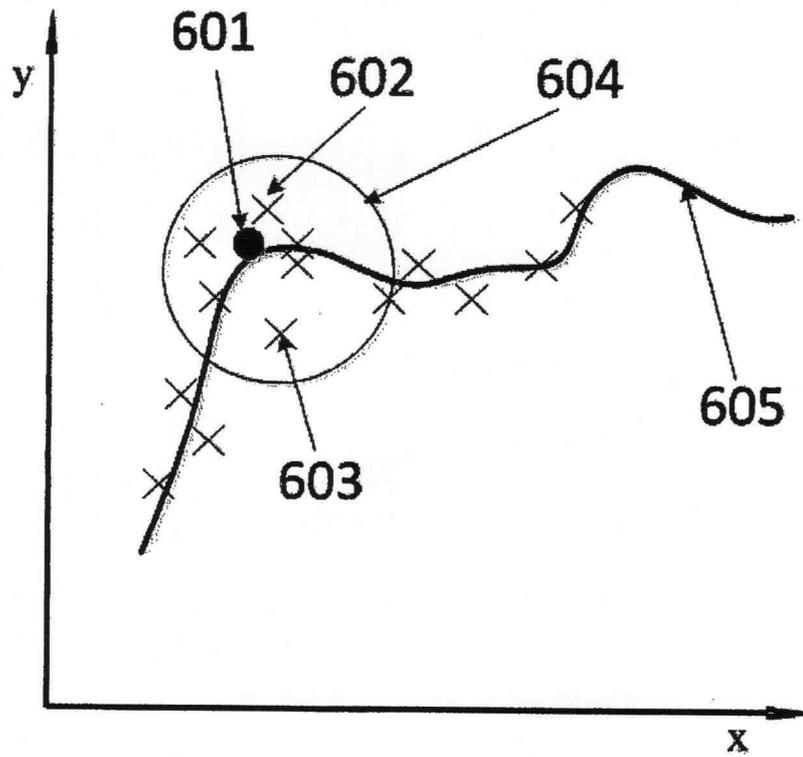


FIG. 6

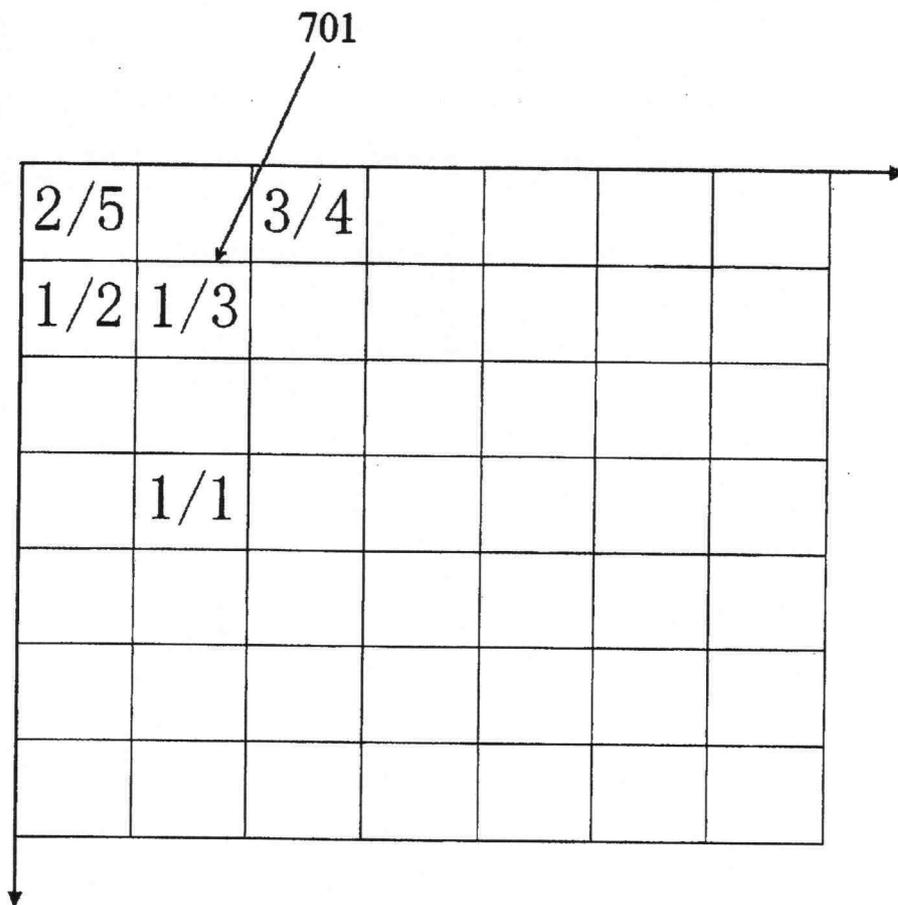


FIG. 7

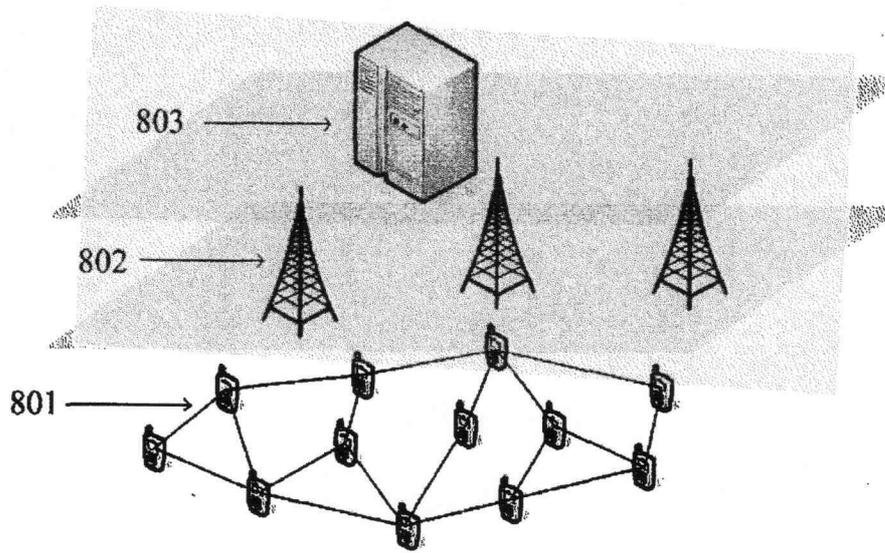


FIG. 8