(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 735 006**

(51) Int. CI.:

H04L 12/24 (2006.01)
H04L 12/707 (2013.01)
H04L 12/715 (2013.01)
H04L 12/721 (2013.01)
H04L 12/703 (2013.01)
H04L 12/753 (2013.01)
H04W 16/18 (2009.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(86) Fecha de presentación y número de la solicitud internacional: **18.02.2013 PCT/IB2013/051320**

(87) Fecha y número de publicación internacional: **29.08.2013 WO13124783**

(96) Fecha de presentación y número de la solicitud europea: **18.02.2013 E 13716052 (9)**

(97) Fecha y número de publicación de la concesión europea: **10.04.2019 EP 2817928**

(54) Título: **Colocación del controlador para la falta rápida en la arquitectura dividida**

(30) Prioridad:

22.02.2012 US 201213402732

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
13.12.2019

(73) Titular/es:

TELEFONAKTIEBOLAGET LM ERICSSON (PUBL) (100.0%)
164 83 Stockholm, SE

(72) Inventor/es:

BEHESHTI-ZAVAREH, NEDA;
ZHANG, YING y
HALPERN, JOEL

(74) Agente/Representante:

LINAGE GONZÁLEZ, Rafael

DESCRIPCIÓN

Colocación del controlador para la falta rápida en la arquitectura dividida

Campo de la invención

Las realizaciones de la invención se refieren a la organización y diseño de redes. Específicamente, las realizaciones de la invención se refieren a un método y sistema para determinar la colocación del controlador para conmutadores en una red de arquitectura dividida con un control desacoplado del reenvío.

Antecedentes

Un diseño de red de arquitectura dividida introduce una separación entre los componentes de control y reenvío de una red. Entre los casos de uso de dicha arquitectura se encuentran el dominio de acceso/agregación de redes de nivel de portadora, retorno móvil, computación en la nube y soporte multicapa (L3 & L2 & L1, OTN, WDM), centros de datos, todos los cuales se encuentran entre los principales bloques de construcción de una arquitectura de red. Por lo tanto, el diseño adecuado, la gestión y la optimización del rendimiento de estas redes son de gran importancia.

A diferencia de la arquitectura de red tradicional, que integra tanto el reenvío (datos) como los planos de control en la misma caja (elemento de red), una red de arquitectura dividida desacopla estos dos planos y ejecuta el plano de control en servidores que pueden estar en diferentes ubicaciones físicas de los elementos de reenvío (conmutadores). El uso de una arquitectura dividida en una red permite la simplificación de los conmutadores que implementan el plano de reenvío y cambia la inteligencia de la red a un número de controladores que supervisan los conmutadores.

El acoplamiento ajustado de los planos de reenvío y control en una arquitectura tradicional generalmente resulta en un plano de control demasiado complicado y una gestión de red compleja. Se sabe que esto crea una gran carga y una alta barrera para los nuevos protocolos y desarrollos tecnológicos. A pesar de la rápida mejora de las velocidades de línea, las densidades de puerto y el rendimiento, los mecanismos del plano de control de red han avanzado a un ritmo mucho más lento que los mecanismos del plano de reenvío.

En una red de arquitectura dividida, los controladores recopilan información de los conmutadores y calculan y distribuyen las decisiones de reenvío adecuadas a los conmutadores. Los controladores y conmutadores usan un protocolo para comunicarse e intercambiar información. Un ejemplo de dicho protocolo es OpenFlow (véase www.openflow.org), que proporciona un método abierto y estándar para que un conmutador se comunique con un controlador, y ha atraído un interés significativo tanto de los académicos como de la industria. La figura 1 es un diagrama que muestra una descripción general de la interfaz OpenFlow entre un conmutador y un controlador. La tabla de reenvío en un conmutador OpenFlow se llena con entradas que consisten en: una regla que define coincidencias para campos en encabezados de paquetes; una acción asociada a la coincidencia de flujo; y una recopilación de estadísticas sobre el flujo.

Cuando un paquete entrante coincide con una regla particular, las acciones asociadas se realizan en el paquete. Una regla contiene campos clave de varios encabezados en la pila de protocolos, por ejemplo, direcciones MAC de Ethernet, dirección IP, protocolo IP, números de puerto TCP/UDP, así como el número de puerto entrante. Para definir un flujo, se pueden usar todos los campos coincidentes disponibles. Pero también es posible restringir la regla coincidente a un subconjunto de los campos disponibles mediante el uso de comodines para los campos no deseados.

La plataforma de control desacoplada de la arquitectura dividida facilita la tarea de modificar la lógica de control de la red y proporciona una interfaz programática sobre la cual los desarrolladores pueden construir una amplia variedad de nuevos protocolos y aplicaciones de gestión. En este modelo, los datos y planos de control pueden evolucionar y escalar de forma independiente, mientras que el costo de los elementos del plano de datos se reduce.

Los siguientes documentos divulgan técnicas de antecedentes tecnológicos adicionales en relación con redes de arquitectura dividida:

D1 YING ZHANG ET AL.: “Sobre resistencia de redes de arquitectura”, CONFERENCIA DE TELECOMUNICACIONES GLOBAL, (GLOBECOM 2011), 2011, IEEE, IEEE, 5 de diciembre de 2011 (.201105-12-2011), páginas 1-6,

XP032119688,

DOI: 10.1109/GLOCOM.2001.6134496

ISBN: 978-1-4244-9266-4

D2 Nick McKeown ET AL.: “OpenFlow: permitir la innovación en redes de campo”, 14 de marzo de 2008 (14-03-2008), páginas 1-6, XP055002028,

Recuperado de Internet: URL: http://www.openflow.org/documents/openflow-wp-latest.pdf [recuperado el 05-07-2011]

D3 EP 2552065 A1 (ERICSSON TELEFON AB L M [SE]) 30 de enero de 2013 (30-01-2013)

D4 NEDA BEHESHTI ET AL.: “Conmutación por error rápida para tráfico de control en redes definidas por software”, CONFERENCIA DE COMUNICACIÓN GLOBAL (GLOBECOM), 2012 IEEE, IEEE, 3 de diciembre de 2012, (03-12-2012), páginas 2665-2670, XP032375076,

DOI: 10-1109/GLOBOCOM.2012-6503519

ISBN: 978-1-4673-0920-2

El documento D1 explica la colocación del controlador y la resistencia de la red con respecto a los algoritmos llamados “algoritmo de corte mínimo” y “algoritmo codicioso”. La topología de la red se mapea a un gráfico y una función de coste es minimizada para encontrar una colocación óptima del controlador de tal manera que se protegen tantos conmutadores como sea posible.

El documento D2 explica el protocolo OpenFlow para el uso de redes de arquitectura dividida.

El documento D3 también explica el concepto de considerar la resistencia de red para calcular la métrica para los conmutadores. Una colocación óptima del controlador es determinada también con un algoritmo.

En el documento D4 un “algoritmo óptimo” y un “algoritmo codicioso” son explicados para la colocación del controlador.

Sumario

Las realizaciones de la invención incluyen un método implementado por un sistema de diseño de topología de red, como se reivindica en la reivindicación independiente 1.

Las realizaciones incluyen una red con una arquitectura dividida, como se reivindica en la reivindicación independiente 9.

Las realizaciones incluyen un sistema informático como se reivindica en la reivindicación independiente 12.

Se proporcionan aspectos adicionales en las reivindicaciones independientes

Breve descripción de los dibujos

La presente invención se ilustra a modo de ejemplo, y no a modo de limitación, en las figuras de los dibujos adjuntos, en los que las referencias similares indican elementos similares. Se debe tener en cuenta que las diferentes referencias a "una" o "una sola" realización en esta divulgación no son necesariamente a la misma realización, y tales referencias significan al menos una. Además, cuando un rasgo, estructura o característica particular se describe en relación con una realización, se afirma que es del conocimiento de un experto en la técnica afectar dicho rasgo, estructura o característica en relación con otras realizaciones, tanto s se describe explícitamente como si no.

La figura 1 es un diagrama de una realización de una arquitectura de ejemplo para una red OpenFlow.

Las figuras 2A y 2B son diagramas de una realización de una red de arquitectura dividida que contiene conmutadores tanto protegidos como no protegidos, cada figura ilustra un mecanismo de protección separado.

La figura 3 es un diagrama de una realización de un sistema de diseño acoplado a una red con colocación del controlador optimizada.

La figura 4 es un diagrama de flujo de una realización de un proceso de optimización de colocación del controlador.

La figura 5 es un diagrama de flujo de una realización de un proceso de colocación del controlador óptimo.

La figura 6 es un diagrama de flujo de una realización de un proceso de colocación del controlador "codicioso".

Descripción detallada

En la siguiente descripción, se exponen numerosos detalles específicos. Sin embargo, se entiende que las realizaciones de la invención pueden ponerse en práctica sin estos detalles específicos. En otros casos, los circuitos, estructuras y técnicas bien conocidos no se han mostrado en detalle para no dificultar la comprensión de esta descripción. Sin embargo, un experto en la técnica apreciará que la invención puede ponerse en práctica sin tales detalles específicos. Los expertos en la técnica, con las descripciones incluidas, podrán implementar la funcionalidad apropiada sin experimentación excesiva.

Las operaciones de los diagramas de flujo se describirán con referencia a las realizaciones de ejemplo de diagramas. Sin embargo, debe entenderse que las operaciones de los diagramas de flujo pueden realizarse mediante realizaciones de la invención distintas de las discutidas con referencia a los diagramas, y las realizaciones explicadas con referencia a diagramas pueden realizar operaciones diferentes a las explicadas con referencia a los diagramas de flujo.

Las técnicas mostradas en las figuras se pueden implementar usando el código y los datos almacenados y ejecutados en uno o más dispositivos electrónicos (por ejemplo, una estación de extremo, un elemento de red, un servidor o dispositivos electrónicos similares). Tales dispositivos electrónicos almacenan y comunican (internamente y/o con otros dispositivos electrónicos a través de una red) el código y los datos usando medios no transitorios legibles por máquina o legibles por ordenador, tales como medios de almacenamiento no transitorios legibles por máquina o legibles por ordenador (por ejemplo, discos magnéticos, discos ópticos, memoria de acceso aleatorio, memoria de solo lectura, dispositivos de memoria flash y memoria de cambio de fase). Además, tales dispositivos electrónicos incluyen típicamente un conjunto de uno o más procesadores acoplados a uno o más componentes, como uno o más dispositivos de almacenamiento, dispositivos de entrada/salida del usuario (por ejemplo, un teclado, una pantalla táctil y/o una pantalla), y conexiones de red. El acoplamiento del conjunto de procesadores y otros componentes se realiza típicamente a través de uno o más buses y puentes (también denominados controladores de bus). Los dispositivos de almacenamiento representan uno o más medios de almacenamiento no transitorios legibles por máquina o legibles por ordenador y medios de comunicación no transitorios legibles por máquina o legibles por ordenador. Por lo tanto, el dispositivo de almacenamiento de un dispositivo electrónico dado típicamente almacena código y/o datos para su ejecución en el conjunto de uno o más procesadores de ese dispositivo electrónico. Por supuesto, una o más partes de una realización de la invención pueden implementarse usando diferentes combinaciones de software, firmware y/o hardware.

Como se usa en el presente documento, un elemento de red (por ejemplo, un enrutador, un conmutador, un puente o un dispositivo de red similar) es una pieza de equipo de red, que incluye hardware y software que interconecta de forma comunicativa a otros equipos en la red (por ejemplo, otros elementos de red, estaciones de extremo o dispositivos de red similares). Algunos elementos de red son "elementos de red de servicios múltiples" que proporcionan soporte para múltiples funciones de red (por ejemplo, enrutamiento, conexión en puente, conmutación, agregación de Capa 2, control de borde de sesión, multidifusión y/o gestión de abonados), y/o proporcionan soporte para múltiples servicios de aplicación (por ejemplo, recolección de datos). Las realizaciones descritas en el presente documento usan el ejemplo de elemento de red en forma de un conmutador. Sin embargo, las realizaciones no están limitadas a los conmutadores y son aplicables a otros tipos de elementos de red.

Tal como se usa en el presente documento, la resistencia es la capacidad de proporcionar y mantener un nivel aceptable de servicio ante fallos y desafíos para el funcionamiento normal. Como se usa en el presente documento, la probabilidad de fallo es la frecuencia con la que falla un sistema o componente diseñado, expresado como el número de fallos por hora, o la probabilidad de que cada nodo falle a largo plazo.

Al evaluar el diseño de una red, la resistencia de la red es un factor importante, ya que un fallo de unos pocos milisegundos puede fácilmente ocasionar pérdidas de datos de terabyte en enlaces de alta velocidad. En las redes tradicionales, donde tanto el control como los paquetes de datos se transmiten en el mismo enlace, el control y la información de los datos se ven igualmente afectados cuando hay un fallo. El trabajo existente en la resistencia de la red, por lo tanto, ha asumido un modelo de control en banda, lo que significa que el plano de control y el plano de datos tienen las mismas propiedades de resistencia. Sin embargo, este modelo no es aplicable a las redes de arquitectura dividida.

Un fallo de enlace indica que el tráfico que recorre un enlace ya no se puede transferir a través del enlace. El fallo puede ser un enlace entre dos conmutadores o un enlace entre un controlador y el conmutador al que se conecta. En la mayoría de los casos, estos enlaces fallan de forma independiente.

Un fallo de conmutación indica que el elemento de red correspondiente no puede originar, responder ni reenviar ningún paquete. Los fallos de conmutador pueden ser causados por errores de software, fallos de hardware, configuraciones erróneas y problemas similares. En la mayoría de los casos, estos conmutadores fallan independientemente.

Los casos especiales de fallo incluyen la pérdida de conectividad entre un conmutador y un controlador. Un conmutador puede perder la conectividad con su controlador debido a fallos en los enlaces intermedios o nodos a lo largo de la ruta entre el conmutador y el controlador. En una realización, siempre que un conmutador no pueda

comunicarse con su controlador asignado, el conmutador descartará todos los paquetes en el plano de reenvío gestionado por el controlador, incluso aunque la ruta en el plano de reenvío aún sea válida. En otras realizaciones, un subconjunto del tráfico puede reenviarse en un plano de reenvío o una funcionalidad limitada similar puede continuar por un tiempo limitado hasta que se restablezca una conexión con un controlador asignado u otro controlador. Por lo tanto, esto puede ser considerado como un caso especial de fallo de conmutador.

Los paquetes de control en redes de arquitectura dividida se pueden transmitir en diferentes rutas del paquete de datos (o incluso en una red separada). Por lo tanto, la confiabilidad del plano de control en estas redes ya no está directamente vinculada con la del plano de reenvío. Sin embargo, la desconexión entre el controlador y el plano de reenvío en la arquitectura dividida podría deshabilitar el plano de reenvío; cuando un conmutador se desconecta de su controlador, no puede recibir ninguna instrucción sobre cómo reenviar nuevos flujos y queda prácticamente fuera de línea.

En una realización de una red de arquitectura dividida, cada conmutador está preprogramado con una ruta para llegar al controlador. Al producirse un fallo de enlace o nodo, el conmutador confía en el controlador para detectar tal fallo y volver a calcular la nueva ruta para el conmutador. Sin embargo, el manejo de todos los fallos por parte del controlador podría resultar en grandes retrasos en la red. En otra realización, la configuración previa de una ruta de respaldo y/o una tunelización hacia un conmutador intermedio se usa para restablecer la comunicación con un controlador, de modo que si el enlace de salida primario no funciona correctamente, se podría usar el enlace de salida de respaldo (secundario) o una encapsulación del tráfico de control a través de un túnel a un conmutador intermedio.

Cuando un conmutador detecta un fallo en su enlace saliente o su nodo ascendente inmediato, inmediatamente cambia su camino hacia el controlador y usa la ruta de respaldo, es decir, la interfaz saliente, preprogramada en el conmutador para volver a conectarse al controlador. En la alternativa, el conmutador detecta el fallo y encapsula el tráfico de control para la transmisión a través de un túnel a un conmutador intermedio que desencapsula el tráfico de control y reenvía el tráfico de control al controlador. Esto se lleva a cabo sin necesidad de involucrar al controlador y sin ningún efecto en el resto de los caminos de la red y en las conexiones de los nodos descendentes al controlador. En otras palabras, solo habrá un cambio local en la interfaz saliente del conmutador afectado. Todas las demás conexiones en la red permanecerán intactas. Sin tales rutas de respaldo u opciones de encapsulación, la detección de cualquier fallo en los conmutadores o enlaces por parte del controlador debe basarse en algunos mecanismos implícitos, como cuando el controlador no recibe los mensajes de saludo de un conmutador. Esto introduce grandes retrasos en la red para detectar la ubicación exacta del fallo y restablecer las conexiones de conmutador-controlador. Si no se puede configurar una ruta de respaldo o una opción de tunelización para un conmutador, entonces la conexión de conmutador a controlador se interrumpirá en caso de un fallo en la ruta primaria al controlador.

Como se usa en el presente documento, se considera que un conmutador está protegido (en su conexión con el controlador) contra el fallo de su conmutador ascendente inmediato y su enlace saliente si se cumple alguna de las siguientes condiciones: i) el conmutador puede usar un enlace saliente de respaldo para su tráfico de control hacia el controlador, o ii) el conmutador puede enviar su tráfico de control a través de un túnel a otro conmutador (intermedio) y desde allí al controlador.

Cuando hay un fallo en el enlace saliente o en el nodo ascendente inmediato de un conmutador protegido, el conmutador puede usar el enlace saliente de respaldo (si la condición i es válida) para volver a conectarse al controlador. En la alternativa (si se cumple la condición ii), el conmutador puede encapsular el mensaje de control dentro de un mensaje de datos y enviarlo a otro conmutador (intermedio). Cuando el conmutador intermedio recibe este mensaje, desencapsulará el mensaje y lo enviará, como su propio tráfico de control, al controlador.

Si no se cumple ninguna de las dos condiciones anteriores, en caso de un fallo en el enlace saliente o el conmutador ascendente inmediato, la conexión entre el conmutador y el controlador se interrumpirá. El objetivo es minimizar la posibilidad de tal interrupción. El escenario más resistente es, claramente, cuando cada conmutador en la red está protegido. Pero si ese no es el caso, entonces se requiere cierta optimización para minimizar el riesgo de interrupción del tráfico de control.

Al usar este esquema de protección en una red de arquitectura dividida, es importante colocar el controlador de modo que sea menos probable que se interrumpa la conexión entre el plano de control y el plano de reenvío. Una buena selección de la ubicación del controlador debe dar como resultado rutas confiables desde los conmutadores al controlador, en el sentido de que una gran cantidad de conmutadores deben tener rutas de respaldo al controlador.

Las realizaciones de la invención proporcionan un método y un sistema para evitar las desventajas de la técnica anterior. Las propuestas existentes en el diseño de red de arquitectura dividida asumen ubicaciones fijas para los controladores de red. Si bien ha habido algunas investigaciones sobre los mecanismos de enrutamiento entre los controladores de red y los conmutadores, no se han desarrollado estrategias para elegir la ubicación optimizada para el controlador de red. Como resultado, la colocación del controlador en arquitecturas divididas no tiene en cuenta la posibilidad de desconexión entre un controlador y el plano de reenvío y busca minimizar esta posibilidad.

Además, los esquemas para redes de arquitectura dividida con múltiples controladores se centran en la partición de la red y la asignación de un controlador a cada partición de tal manera que los conmutadores dentro de cada partición estén bien conectados. Esto no aborda la búsqueda de una ubicación óptima para un controlador en una red determinada sin partición. Los esquemas para colocar un solo controlador en una red de arquitectura dividida pueden colocar al controlador en un nodo que maximice la resistencia de la conexión entre el controlador y los conmutadores, sin embargo, estos esquemas se basan en una definición de protección restringida. En tales esquemas, un conmutador protegido es un conmutador con un enlace saliente de respaldo y no considera la posibilidad de enviar el tráfico de control a través de un túnel a otro conmutador y desde allí al controlador.

Las realizaciones de la invención superan estas desventajas de la técnica anterior. Las realizaciones de la invención colocan un solo controlador en un área de arquitectura dividida, en una ubicación seleccionada para optimizar la resistencia de la conexión entre el controlador y los conmutadores en esa área. No se hacen suposiciones sobre cómo se realiza la partición de las áreas de arquitectura dividida. La partición, en su caso, puede basarse en cualquier métrica arbitraria, como las restricciones geográficas. Las realizaciones de la invención abarcan dos procesos de ejemplo (es decir, un proceso óptimo y un proceso codicioso) para elegir la ubicación del controlador para optimizar la resistencia de la conexión entre el controlador y los conmutadores, es decir, para maximizar el número de conmutadores con rutas de respaldo preconfiguradas al controlador a través de enlaces de respaldo directos o mediante el tráfico de control de tunelización a un elemento de red intermedio que no se encuentra en sentido descendente desde el punto de fallo.

Las realizaciones soportan una definición más general para un conmutador protegido. Si no hay una interfaz saliente de respaldo para un conmutador, el conmutador todavía se considera protegido si puede enviar su tráfico de control a otro conmutador (intermedio) y desde allí al controlador. En este caso, el conmutador encapsula el mensaje de control dentro de un mensaje de datos al conmutador intermedio. Cuando el conmutador intermedio recibe este mensaje, desencapsulará el mensaje y lo enviará (como su propio tráfico de control) al controlador. Este mecanismo de protección alternativo se refiere en el presente documento como protección basada en tunelización, y el término tunelización se refiere al proceso de encapsular el mensaje de tráfico dentro de un mensaje de datos, enviarlo al conmutador intermedio y, finalmente, desencapsularlo en el conmutador intermedio. Usando esta definición más general de protección, las realizaciones incluyen procesos y sistemas para colocar de manera óptima el controlador en la red de manera que se maximice la resistencia.

Ubicación del controlador de red

La resistencia de la conexión entre el plano de control y el plano de reenvío es de gran importancia en las redes de arquitectura dividida. Si se interrumpe esta conexión, entonces el plano de reenvío no sabrá cómo reenviar nuevos flujos (es decir, esos flujos sin reglas existentes en los conmutadores) y perderá su funcionalidad de reenvío. Las realizaciones de la invención proporcionan un proceso para decidir dónde colocar el controlador de arquitectura dividida, de manera que es menos probable que se interrumpa esta conexión (entre el plano de control y el plano de reenvío). Dada una topología de red, el proceso busca elegir el nodo correcto en la red para ubicar el controlador en ese nodo. Una buena selección de la ubicación del controlador de una red debe dar como resultado rutas confiables desde los conmutadores al controlador, en el sentido de que cada conmutador debe tener una ruta de respaldo (secundaria) al controlador o una protección basada en tunelización que no se verá afectada por el mismo fallo, en caso de que su ruta principal falle, esta ruta de respaldo puede ser un enlace directo entre el conmutador que detecta el fallo y otro conmutador en la red que permanece en comunicación con el controlador o una protección basada en tunelización en forma de un enlace indirecto entre el conmutador que detecta el fallo y un conmutador intermedio sobre un túnel donde el túnel recorre al menos un conmutador descendente.

Métrica de protección

Para evaluar diferentes estrategias de colocación del controlador en una red (y para desarrollar una política para elegir una buena ubicación), se utiliza una métrica de protección, que se basa en la protección de nodos. Esta métrica se aplica a la arquitectura dividida para evaluar la resistencia de la red frente a fallos de enlace, como se define anteriormente y se explica con más detalle en el presente documento a continuación.

Los fallos transitorios ocurren con relativa frecuencia incluso en redes de protocolo de Internet (IP) bien gestionadas. Sin embargo, se espera que el servicio de red esté siempre disponible con la creciente demanda de entrega de servicios críticos. Con los altos requisitos de confiabilidad de la red, las realizaciones de la invención buscan mejorar la resistencia de la conectividad entre el controlador y los conmutadores en una red de arquitectura dividida.

Entorno de red

Las realizaciones de la invención proporcionan un proceso en el que el reenvío de paquetes de datos se reanuda después de un fallo tan pronto como sea posible. Los protocolos de pasarela interior (IGP) existentes, como abrir primero la ruta más corta (OSPF) y el sistema intermedio al sistema intermedio (IS-IS), típicamente tardan varios segundos en converger, lo que no alcanza un nivel de recuperación de fallos inferior a 50 ms. que se espera para la confiabilidad de la red. El controlador podría detectar los fallos en los conmutadores o enlaces usando algunos

mecanismos implícitos, por ejemplo, cuando el controlador no recibe mensajes de saludo desde un conmutador. Sin embargo, este método también introducirá un gran retraso en la red para la detección de fallos y la restauración del servicio.

En una realización, la decisión de conmutación de protección se realiza localmente y está predeterminada por el controlador (es decir, en el elemento de red que detecta el fallo). Esto es diferente del escenario en una red tradicional, porque el elemento de red no tiene una topología completa de la red. El elemento de red es solo un simple conmutador en el plano de reenvío y solo recibe reglas de reenvío desde el controlador. Cuando se pierde la conectividad con el controlador, el conmutador debe tomar la decisión de realizar una conmutación por error de forma independiente sin recibir instrucciones del controlador. En otras palabras, solo habrá un cambio local en la interfaz saliente del conmutador afectado. Todas las demás conexiones en la red permanecerán intactas. De esta manera, el proceso mantiene el elemento de reenvío, es decir, el conmutador, lo más simple posible.

En una realización, el controlador está en la misma red física que los conmutadores. Es decir, la infraestructura existente de la red de arquitectura dividida (enlaces y conmutadores existentes) se usa para conectar el controlador a todos los conmutadores de la red, en lugar de usar una infraestructura separada para conectar los planos de control y reenvío.

Como se usa en el presente documento, una red de conmutadores se indica mediante un gráfico G = (V, E), donde V es el conjunto de nodos (conmutadores y el controlador) en la red y E es el conjunto de bordes bidireccionales (enlaces) entre nodos. Se asocia un costo a cada enlace en la red. Basándose en los costos de enlace asignados, los caminos de ruta más corta se calculan entre dos nodos cualesquiera en la red. Se supone que el costo en cada enlace se aplica a ambas direcciones del enlace. También se supone que no hay equilibrio de carga en el tráfico de control enviado entre los conmutadores y el controlador. Por lo tanto, cada nodo tiene solo una ruta para llegar al controlador. En otras palabras, el tráfico de control se envía desde y hacia el controlador a través de un árbol, enraizado en el controlador, al que se hará referencia en el presente documento como un árbol de enrutamiento del controlador. Este árbol de enrutamiento cubre todos los nodos de la red y un subconjunto de los bordes. El mismo árbol de enrutamiento se usa para las comunicaciones entre el controlador y los conmutadores en ambas direcciones.

Con una ubicación de controlador determinada, cualquier protocolo de enrutamiento de ruta más corta forma un árbol T, enraizado en el nodo del controlador, que cubre todos los nodos y un subconjunto de los bordes. Como se mencionó anteriormente, este árbol se conoce como el árbol de enrutamiento del controlador. Las figuras 2A y 2B muestran una red y su árbol de enrutamiento del controlador. En estas figuras, las líneas discontinuas muestran todos los enlaces en la red, y las líneas continuas muestran los enlaces usados en el árbol de enrutamiento del controlador. Cada nodo puede llegar al controlador enviando su tráfico de control a lo largo de las rutas en el árbol de enrutamiento del controlador. En estos ejemplos, ambas direcciones de cada enlace tienen el mismo costo y, por lo tanto, el mismo árbol de enrutamiento se usará para las comunicaciones entre el controlador y los conmutadores en ambas direcciones.

En el árbol de enrutamiento del controlador T, el nodo u es un nodo ascendente del nodo v si hay una ruta en T desde el nodo v al nodo u hacia el controlador. El nodo u se denomina nodo descendente del nodo v si hay una ruta en T desde el nodo u al nodo v hacia el controlador. En el ejemplo de redes ilustradas en las figuras 2A y 2B, por ejemplo, el nodo S4 es un nodo ascendente de los nodos S7 y S8, y estos dos nodos son nodos descendentes del nodo S4. En el árbol de enrutamiento del controlador, el padre de un nodo es su nodo ascendente inmediato y los hijos de un nodo son sus nodos descendentes inmediatos. Debido a la estructura de árbol asumida, cada nodo tiene solo un nodo ascendente inmediato en T. En el ejemplo y en las realizaciones del proceso de colocación del controlador, se supone que no hay equilibrio de carga en el tráfico de control enviado desde los conmutadores al controlador. Es decir, asumimos que cada nodo en la red tiene solo un nodo ascendente inmediato en T. Los símbolos introducidos en el presente documento (por ejemplo, G, T, u y v) se usan en el presente documento a continuación para representar estos conceptos por razones de claridad y precisión.

Fallos de nodo y enlace

Como se explicó anteriormente en el presente documento, se considera que un conmutador está protegido (en su conexión con el controlador) contra el fallo de su conmutador ascendente inmediato y su enlace saliente si el conmutador puede:

i) Usar un enlace saliente de respaldo para controlar el tráfico hacia el controlador; o

ii) Enviar su tráfico de control a través de un túnel a otro conmutador (intermedio) y desde allí al controlador.

Por ejemplo, un conmutador protegido que detecta un fallo en su enlace saliente o su nodo ascendente inmediato si la condición (i) se cumple, tan pronto como se detecte el fallo, cambiará inmediatamente su camino hacia el controlador y usará el enlace saliente de respaldo para reconectarse al controlador. Si se cumple la condición (ii), entonces el conmutador puede encapsular el mensaje de control dentro de un mensaje de datos al conmutador

intermedio. Cuando el conmutador intermedio recibe este mensaje, desencapsulará el mensaje y lo enviará (como su propio tráfico de control) al controlador. En ambos casos, el redireccionamiento del tráfico de control tiene lugar sin ningún impacto en el resto de las conexiones de otros conmutadores al controlador. En otras palabras, solo habrá un cambio local en la interfaz saliente del conmutador afectado. Todas las demás conexiones en la red permanecerán intactas. En una realización, el conmutador puede llevar a cabo cualquiera de estos procesos de conmutación por error (es decir, aquellos vinculados a la condición (i) o (ii)) automáticamente sin la participación del controlador.

Si ninguna de estas dos condiciones se cumple, entonces, en caso de un fallo en la ruta primaria al controlador, la conexión entre el conmutador y el controlador se interrumpirá. El proceso de colocación del controlador y el sistema descritos en la presente invención están diseñados para minimizar la posibilidad de tal interrupción. La configuración más resistente de la red es, claramente, cuando todos los conmutadores de la red están protegidos. Pero si esa configuración no es posible, entonces se requiere cierta optimización de la colocación del controlador para minimizar el riesgo de interrupción del tráfico de control entre el controlador y los conmutadores en la red.

Para aquellos conmutadores que están conectados directamente al controlador, la protección del nodo ascendente no está definida ni cuantificada, ya que el nodo ascendente inmediato es el controlador. En las redes de arquitectura dividida en las que se implementan las herramientas tradicionales de gestión de fallos, no existe un mecanismo de señalización extendido para que un nodo informe a sus nodos descendentes de un fallo. Por lo tanto, si un conmutador se desconecta del controlador, todos sus nodos descendentes también se desconectarán, incluso si ellos mismos están protegidos contra sus enlaces salientes o fallos inmediatos de los nodos ascendentes. Esto significa que al evaluar la resistencia de las redes, se debe asignar más importancia a los nodos más cercanos al controlador (que es la raíz del árbol de enrutamiento del controlador). Para representar estas facetas de la red que afectan la resistencia de la red, se definen ponderaciones para cada nodo que se basan en el número de sus nodos descendentes.

Una ponderación de un árbol de enrutamiento puede definirse como la suma de las ponderaciones de todos sus nodos no protegidos. Esta ponderación se puede usar para medir la "desprotección" o la resistencia de la red para una posición de controlador asociada. Para un árbol de enrutamiento T dado, esta ponderación del árbol de enrutamiento se puede describir o representar con la "ponderación (T)", que debe minimizarse para maximizar la resistencia de la red.

Las figuras 2A y 2B muestra un ejemplo de redes y dos escenarios de fallo. Las líneas continuas entre los conmutadores y el controlador en estas figuras muestran el árbol de ruta más corta entre el controlador y los conmutadores. Si no hay fallos en la red, el tráfico de control se enviará a/desde el controlador en este árbol representado por las líneas continuas.

Por ejemplo, el conmutador S4 en esta red está conectado al controlador a través de su padre ascendente S1. En ambos escenarios mostrados en las figuras 2A y 2B, el conmutador S4 está protegido. Esto se debe a que en caso de fallo en el conmutador ascendente inmediato S1 o el enlace que conecta S4 y S1, todavía hay una ruta de respaldo para que el tráfico de control del conmutador S1 llegue al controlador. En el caso ilustrado en la figura 2A, hay un enlace entre S4 y S5 representado por la línea de puntos. Este enlace no forma parte del árbol de enrutamiento, por lo que este enlace se puede configurar en el conmutador S4 como un enlace saliente de respaldo para el tráfico de control. Por lo tanto, si S4 detecta un fallo en el enlace saliente primario entre los conmutadores S4 y S1 o en el conmutador ascendente S1, entonces el conmutador S4 puede usar el enlace saliente de respaldo entre los conmutadores S4 y S5.

En el caso ilustrado en la figura 2B, no hay un enlace que conecte S4 a otro conmutador que pueda usarse como enlace de respaldo. Se debe tener en cuenta que ninguno de los enlaces que conectan S4 con sus hijos (conmutadores S6 y S8) se pueden usar como un enlace saliente de respaldo para el tráfico de control, ya que no tienen una ruta en el árbol de enrutamiento al controlador que no pasa por el enlace fallido o el conmutador fallido (es decir, el enlace entre los conmutadores S4 y S1 o el conmutador S1). En este caso, sin embargo, hay un enlace entre los conmutadores S8 y S9. Aquí, el conmutador S4 puede hacer un túnel desde el conmutador S8 al conmutador S9 (encapsulando el tráfico de control con el conmutador S9 como destino). Cuando el conmutador S9 recibe y desencapsula este tráfico, puede enviar el tráfico al controlador (como su propio tráfico de control) en la ruta S9-S5-S2-controlador. Cabe señalar que esta ruta no pasa a través de S4 y S1, evitando así el enlace o conmutador fallido en este ejemplo. En otras palabras, el controlador ha seleccionado un conmutador intermedio cuya ruta hacia el controlador no se ve afectada por el fallo de conmutador S1 o el enlace entre los conmutadores S4 y S1.

Evaluación del estado de protección de un conmutador

En una realización, cada conmutador S en una red de arquitectura dividida puede tener evaluado su estado de protección. Tal como se usa en el presente documento, 'padre (S)' indica el conmutador ascendente inmediato del conmutador S, y 'sentido descendente (S)' indica todos los conmutadores descendentes del conmutador S (es decir, sus hijos e hijos de hijos y así sucesivamente). Cada conmutador S en una red dada está protegido de acuerdo con

nuestra definición anterior si, y solo si, existen los conmutadores A y B en la red, de manera tal que se use la notación de teoría de conjuntos estándar:

1. A está en {S} U sentido descendente (S), es decir, A es S o uno de los nodos descendentes del conmutador S.

2. B no está en el sentido descendente (padre (S)).

3. Existe un enlace entre A y B, que no forma parte del árbol de enrutamiento del controlador.

Si se cumplen las tres condiciones anteriores, en caso de fallo, el conmutador S puede enviar su tráfico de control a través de un túnel al conmutador B y de allí al controlador. Si el conmutador A es S, el conmutador S puede usar el enlace S-B como enlace saliente de respaldo para el tráfico de control; por lo tanto, no hay necesidad de tunelización en este caso especial. Básicamente, las condiciones anteriores garantizan que el tráfico de control pueda enviarse a través de un subárbol diferente al enraizado en el padre del nodo S. Es decir, el tráfico podría omitir el conmutador/enlace fallido.

Dado que el árbol de enrutamiento del controlador es el árbol de ruta más corta, las tres condiciones anteriores también garantizan que la ruta desde el conmutador B al controlador no pase por S y su nodo ascendente inmediato (padre). Por lo tanto, el controlador S-B de ruta podría usarse cuando el conmutador S detecta un fallo (ya sea en su nodo ascendente inmediato o en el enlace que conecta S a su nodo ascendente inmediato).

Volviendo a los ejemplos de las figuras 2A y 2B, los conmutadores A = S4 y B = S5 en la figura 2A satisfacen las tres condiciones anteriores, y en la figura 2B, los conmutadores A = S8 y B = S9 satisfacen estas condiciones.

Implementación de protección usando OpenFlow

En una realización, el proceso de colocación del controlador puede aplicarse a cualquier implementación de una red de arquitectura dividida. La tabla de reenvío en un conmutador OpenFlow, por ejemplo, se rellena con entradas que consisten en una regla que define coincidencias para campos en encabezados de paquetes, un conjunto de acciones asociadas con la coincidencia de flujo y una colección de estadísticas sobre el flujo. La versión 1.1 de la especificación OpenFlow introduce un método para permitir que un único activador de coincidencia de flujo se envíe en más de uno de los puertos del conmutador. La conmutación por error rápida es uno de estos métodos. Usando este método, el conmutador ejecuta el primer conjunto de acciones en vivo. Cada conjunto de acciones está asociado con un puerto especial que controla su vida. El método rápido de conmutación por error de OpenFlow permite que el conmutador cambie el reenvío sin requerir un viaje de ida y vuelta al controlador.

Proceso de colocación del controlador

La protección de los nodos en una red depende tanto de la selección de las rutas primarias (para una ubicación del controlador dada) como de la elección de la ubicación del controlador. Como se establece a continuación, se define una política de enrutamiento general que, para cada elección de la ubicación del controlador, selecciona las rutas principales en la red para llegar al controlador. Esta selección podría basarse en cualquier métrica deseada, por ejemplo, métricas de rendimiento como retraso o carga. También se analiza lo que incluye una búsqueda exhaustiva para encontrar la mejor ubicación para estas rutas primarias seleccionadas arbitrariamente.

Arquitectura del sistema de diseño y red de ejemplo con ubicación de controlador optimizada

La figura 3 es un diagrama de una realización de un sistema de diseño acoplado a una red con una colocación del controlador optimizada. El diagrama proporciona una ilustración de un ejemplo de sistema 301 de diseño de red para ejecutar la herramienta del sistema de diseño de red. El sistema 301 de diseño de red puede ser cualquier tipo de dispositivo informático, incluida un ordenador de escritorio, un servidor, un dispositivo informático de mano, un dispositivo de consola, un dispositivo portátil o un dispositivo informático similar. El sistema 301 de diseño de red incluye un conjunto de procesadores 303 para ejecutar los componentes de la herramienta del sistema de diseño de red que incluye un módulo 305 de gráfico de topología, un módulo 307 de colocación del controlador y componentes similares. En otras realizaciones, cualquiera o todos estos módulos pueden implementarse como un conjunto de módulos o dispositivos de hardware. El procesador 303 también puede ejecutar un módulo 309 de gestión de red para comunicarse y/o gestionar la red de arquitectura dividida.

El módulo 305 de gráfico de topología puede convertir una topología de red en un gráfico representativo y realizar funciones gráficas en el gráfico representativo para soportar el módulo 307 de colocación del controlador. El módulo 307 de colocación del controlador opera en el gráfico generado por el módulo 305 de gráfico de topología y dirige las operaciones gráficas para implementar un proceso de colocación óptimo o un proceso de colocación "codicioso" para determinar una ubicación para un controlador como se describe más adelante en el presente documento.

El módulo 309 de gestión de red puede comunicarse con el módulo 303 de colocación del controlador y/o el módulo 305 de gráfico de topología para descubrir la topología de la red para un proceso automatizado y/o para implementar

la colocación del controlador en un proceso automatizado. En otras realizaciones, el módulo 307 de colocación del controlador genera un informe o salida similar a un usuario para implementar una organización de red y el módulo 309 de gestión de red puede omitirse.

La red de arquitectura dividida ilustrada es una implementación de ejemplo con una colocación del controlador de ejemplo consistente con la optimización de la colocación del controlador. En el ejemplo, hay un controlador 315 para controlar el dominio o el área de arquitectura dividida que consta de los conmutadores 317. Los conmutadores 317 son gestionados por el controlador 315 usando el árbol 319 de enrutamiento del controlador mostrado como líneas de puntos que conectan los conmutadores 317, donde las líneas continuas 321 son los enlaces entre los conmutadores 317. El proceso para determinar la ubicación del controlador 315 se describe a continuación en el presente documento.

Ubicación optimizada del controlador para una conmutación por error rápida

El proceso general de colocación del controlador se describe con respecto a la figura 4. La entrada del proceso de colocación del controlador es el gráfico de topología de la red G = (V, E), y la salida es el controlador_ubicación, es decir, el nodo de red en el que se ubicará el controlador.

El proceso general de colocación del controlador se inicia haciendo un gráfico la topología de la red de arquitectura dividida (bloque 401). Los nodos y los enlaces entre los nodos se pueden determinar mediante la entrada del administrador, los procesos de descubrimiento automatizados o cualquier combinación de los mismos. El gráfico representa elementos de red (por ejemplo, conmutadores) en la red como nodos en un gráfico con los enlaces de comunicación entre estos elementos de red representados como enlaces o bordes en el gráfico.

Después, el proceso recorre los nodos en el gráfico para calcular una métrica de protección para cada nodo en el gráfico (bloque 403). La métrica de protección como se describe en el presente documento anteriormente y más adelante en el presente documento, mide la resistencia de la red de arquitectura dividida como un grado de protección de fallo de nodo para cada ubicación de controlador posible dentro de la red, es decir, para cada nodo o elemento de red posible en la red que puede hospedar el controlador. La métrica de protección mide la resistencia de la red de arquitectura dividida como un grado de protección de fallo de nodo dentro de la red de arquitectura dividida para una posible colocación del controlador. El grado de protección de fallo de nodo determina un subconjunto de nodos protegidos (es decir, elementos de red protegidos) en el conjunto de nodos (es decir, el conjunto de elementos de red), donde un nodo protegido en el subconjunto de nodos protegidos puede redirigir el tráfico de control a través de un túnel a un nodo intermedio en el gráfico que no esté en sentido descendente del nodo protegido, y donde el túnel recorra al menos un nodo descendente del nodo protegido.

Una vez que se determina la métrica de protección para cada nodo en el gráfico, se selecciona el elemento de red correspondiente al nodo en el gráfico con la mejor métrica de protección (bloque 405). El elemento de red seleccionado se envía luego al administrador de red para la implementación manual o a un módulo de gestión de red para la implementación automatizada o cualquier combinación de los mismos. La selección de un elemento de red mediante este proceso proporciona una estrategia de protección optimizada para la red en su conjunto.

Hay dos procesos de ejemplo más específicos para recorrer el gráfico y determinar la métrica de protección para los nodos en ella. En el primer proceso, un proceso de colocación óptimo, se buscan todas las ubicaciones posibles para el controlador y se elige la que maximiza el número de conmutadores protegidos. En un segundo proceso, un proceso 'codicioso', se realiza un recorrido más rápido y sencillo de los nodos con una evaluación más aproximada.

Colocación del controlador - Proceso de colocación óptima

Una realización del proceso se ilustra a continuación en la Tabla I como pseudocódigo.

Tabla I

```
Proceso de colocación óptima
1.  V = conjunto de todos los nodos en la red; n = |V|
2.  para cada nodo v en V hacer
3.  T = árbol de enrutamiento del controlador enraizado en v
4.  ponderación (T) = 0
5.  para cada nodo u ≠ v hacer
6.  ponderación (u) = 0
7.  Si (u no está protegido) entonces
8.  ponderación (u) = 1 + número de nodos descendentes de u en T
9   fin
10. ponderación (T) = ponderación (T) + ponderación (u);
11. fin
12. fin
```

13. controlador_ubicación= nodo v con ponderación mínima (T)

Como se describe brevemente en la sección anterior, la métrica de protección para cada nodo en una red de gráfico se basa en la ponderación de un árbol arraigado en ese nodo. La ponderación del árbol se calcula donde cada nodo descendente desprotegido en el árbol tiene un árbol enraizado que se agrega a un valor inicial de la ponderación del árbol que se establece en cero (línea 4). Para cada nodo en el árbol que está desprotegido, se asigna una ponderación que se basa en el número de sus nodos descendentes (líneas 7 y 8). Las ponderaciones de cada uno de estos nodos no protegidos se acumulan para calcular la ponderación del árbol (línea 10). Una vez que se generan todas las ponderaciones de los árboles, se selecciona el árbol con la ponderación mínima para la colocación del controlador, ya que proporcionará a la configuración la mayor resistencia debido a que tiene la menor cantidad de nodos desprotegidos cerca del controlador.

Este proceso se describe en relación con el diagrama de flujo de la figura 5. El proceso de colocación óptima es iniciado por el módulo de colocación del controlador en respuesta a la recepción de un gráfico topológico de la red de arquitectura dividida del módulo de gráfico de topología (bloque 501). El proceso luego comienza a iterar a través de cada uno de los nodos en el gráfico (bloque 503). Los nodos se pueden iterar en serie o en paralelo, ya que el orden de evaluación no es importante, ya que cada nodo debe examinarse y se debe generar una métrica de protección.

Para cada nodo en el gráfico, se genera un árbol de enrutamiento del controlador con el nodo dado que sirve como la raíz del árbol (bloque 505). La ponderación de este árbol tiene un valor inicial de cero. Después, para cada uno de estos árboles de enrutamiento, se recorren los nodos dentro de estos árboles (bloque 507). El orden de recorrido de los nodos dentro de los árboles de enrutamiento no es importante y cada uno puede examinarse en paralelo o en serie. Para cada nodo en cada árbol de enrutamiento se da una ponderación inicial de cero (bloque 509). Luego se verifica si el nodo seleccionado actualmente está protegido como se define en el presente documento anteriormente (bloque 51 1). Si el nodo seleccionado actualmente no está protegido, se calcula una ponderación para este nodo (bloque 515). La ponderación se puede calcular por un recuento de la cantidad de nodos que están en sentido descendente del nodo actualmente seleccionado. Este número de nodos descendentes sirve como la ponderación para el nodo seleccionado actualmente en el cálculo de la ponderación del árbol de enrutamiento global. Si el nodo seleccionado actualmente en el árbol de enrutamiento está protegido como se define en el presente documento anteriormente, entonces retiene la ponderación de cero.

A medida que se calcula la ponderación de cada nodo, se suma con la ponderación del árbol actual o la 'ponderación del nodo de raíz actual' (bloque 517). Este proceso de suma se puede hacer de forma iterativa, en cuyo caso se realiza una verificación para determinar si es necesario examinar nodos adicionales en el árbol (bloque 519). El proceso de suma también se puede hacer en un proceso paralelo o un proceso similar.

De manera similar, se realiza una verificación para determinar si todos los nodos de un gráfico se han revisado para determinar la ponderación de su respectivo árbol de enrutamiento del controlador (bloque 521). Esta ponderación del árbol de enrutamiento del controlador puede ser la métrica de protección para el nodo de raíz correspondiente. Una vez que se hayan calculado todas las métricas de protección para todos los nodos en el gráfico, entonces se puede seleccionar el nodo con la mejor métrica de protección (por ejemplo, la ponderación de árbol asociado más bajo o mínimo) para asignar el controlador (bloque 523).

Colocación del controlador - Proceso de colocación codiciosa

Si el tamaño de la red de arquitectura dividida es grande, entonces una búsqueda exhaustiva entre todas las ubicaciones podría volverse muy compleja. En este segundo proceso, presentamos una manera codiciosa de encontrar una ubicación con conexiones ricas entre sus conmutadores conectados directamente. En este proceso, el grado de un nodo v (número de sus vecinos en G) se indica mediante D (v). El proceso comienza seleccionando el nodo v(1) (línea 3), el primer nodo de una lista ordenada de nodos de red, ordenados en un orden de grado decreciente.

Tabla II

Proceso de colocación codicioso

1. V = conjunto de todos los nodos en la red; = |V|;
2. Ordenar los nodos en V de modo que D(v(1)) ≥ D(v(2)) ≥ ... ≥D(v(n))
3. nodo seleccionado ← v(1)
4.
5. para i = 1 a n hacer
6. A = vecinos de v(i) en V
7. D'(v(i)) = número de miembros de A que están conectados a al menos otro miembro de A a través de una ruta que no pasa a través de v(i)
8. si D'(v(i))> D' (nodo seleccionado) entonces nodo seleccionado ← v(i)
9. si (D'(v(i)) == D(v(i)) entonces romper

10. fin
11. controlador_ubicación ← nodo seleccionado

El objetivo de este proceso es encontrar el nodo con el número máximo de vecinos protegidos. Aquí, D'(v) indica el número de vecinos protegidos del nodo v. En la iteración $z^{ésima}$ del proceso, se calcula el número de vecinos protegidos (tal como se define en el presente documento anteriormente) del nodo v(i) (línea 6), y la ubicación del controlador se actualiza al nodo v(i) si se cumple, en términos del número de vecinos protegidos, los nodos buscados anteriormente (líneas 7 y 8). El proceso se detiene cuando encuentra el nodo con el número máximo de vecinos protegidos, que se elegirá como el nodo donde se ubicará el controlador.

La métrica de protección usada en este proceso es el número máximo de vecinos protegidos. Como se explicó anteriormente, los nodos más cercanos al controlador pesan más (que los que están más alejados del controlador), porque si se interrumpe su conexión a la red, todos sus nodos descendentes se verán afectados y desconectados. Por lo tanto, es importante elegir una ubicación para el controlador de modo que sus vecinos, es decir, aquellos conmutadores que están directamente conectados al controlador, estén bien protegidos.

La figura 6 es un diagrama de flujo de una realización del proceso de colocación codicioso. El proceso puede iniciarse recibiendo un gráfico topológico de la red de arquitectura dividida por el módulo de colocación del controlador (bloque 601). El conjunto de nodos se examina para determinar el número de enlaces a los nodos vecinos para cada uno de los nodos del gráfico. Los nodos se ordenan basándose en esta evaluación del número de vecinos (bloque 603). Inicialmente, el nodo con la mayoría de los enlaces vecinos se establece como la ubicación predeterminada o actual para el controlador. Después, el proceso comienza a iterar a través de cada uno de los nodos ordenados comenzando con el nodo con el mayor número de vecinos y avanzando a través de la lista ordenada en orden descendente (bloque 605).

El nodo seleccionado luego se analiza para determinar el número de enlaces a vecinos que están protegidos (bloque 607). Luego se realiza una verificación para comparar el número de enlaces protegidos de este nodo con el número de enlaces protegidos del nodo establecido o inicialmente seleccionado como la ubicación actual (bloque 609). Si el nodo que se está analizando supera el nodo de ubicación actual, entonces, el nodo de ubicación actual se actualiza (bloque 611). El proceso continúa verificando si el número de nodos protegidos del nodo de ubicación actual es menor que el número de vecinos para que se examine el siguiente nodo (bloque 613). Si el número de nodos protegidos excede el siguiente nodo en el número de vecinos de la lista ordenada, entonces el proceso puede completar y generar el nodo seleccionado actual para usarlo como ubicación de colocación del controlador (bloque 615). De lo contrario, el proceso continúa en el siguiente nodo en la lista ordenada.

La resistencia de la red es uno de los factores más importantes en la evaluación de cualquier diseño de red. Un fallo de unos pocos milisegundos puede fácilmente resultar en pérdidas de datos de terabyte en los enlaces de velocidades de transmisión de alta velocidad. Desde la perspectiva de la implementación práctica, estos procesos para la ubicación optimizada del controlador maximizan la resistencia entre el controlador y los conmutadores en la arquitectura dividida. Estos procesos maximizan la resistencia de la red al maximizar el número de conmutadores que están protegidos con rutas de respaldo preconfiguradas o protección basada en tunelización que se encuentran cerca del controlador. En caso de fallos, los elementos de reenvío afectados podrían cambiar inmediatamente a sus rutas de respaldo o caminos basadas en túneles y restaurar sus conexiones con el controlador.

Las realizaciones de la invención pueden proporcionar directrices para que los operadores implementen su red de una manera rentable. Pueden mejorar la resistencia de la red de arquitectura dividida, lo que puede evitar que cientos de miles de flujos se vean afectados por fallos transitorios.

Uso de las redes de arquitectura dividida

Se puede implementar una red de arquitectura dividida para el retorno celular para soportar el reenvío basado en MPLS. En LTE, también se puede implementar en el núcleo móvil para enrutar el tráfico de usuarios entre MME, GW de servicio y PDN-GW. En este caso, el controlador puede implementarse en múltiples sitios o múltiples ubicaciones en un sitio. Los procesos en esta invención se pueden usar para calcular la mejor ubicación para la colocación del controlador.

Cuando coexisten múltiples tecnologías, por ejemplo, GSM, 3G, LTE, pueden compartir las mismas redes de transporte de paquetes. En este ejemplo, se puede usar un conjunto común de controladores para controlar las funciones de conmutación de paquetes para todas las redes juntas. Esta invención se puede usar para determinar la ubicación del controlador para controlar múltiples redes de tecnología.

En la computación en la nube, especialmente las redes de centro de datos, para reducir el costo de la infraestructura de red, se prefiere la arquitectura dividida con un controlador inteligente y un conjunto de conmutadores de bajo costo. En el entorno de red de centro de datos, el proceso de colocación del controlador se puede aplicar para implementar los controladores.

Debe entenderse que la descripción anterior pretende ser ilustrativa y no restrictiva. Muchas otras realizaciones serán evidentes para los expertos en la técnica al leer y comprender la descripción anterior. El alcance de la invención, por lo tanto, debe determinarse con referencia a las reivindicaciones adjuntas.

REIVINDICACIONES

1.- Un método implementado por un sistema de diseño de topología de red, el sistema de diseño de topología de red que incluye un dispositivo de procesamiento, el método para determinar la colocación de un controlador (315) dentro de una red con una arquitectura dividida donde los componentes del plano de control de la red de arquitectura dividida son ejecutados por un controlador (315) y los componentes del plano de control están separados de los componentes del plano de datos de la red de arquitectura dividida, en el que la colocación del controlador se selecciona para minimizar la interrupción de la red de arquitectura dividida causada por un fallo de enlace, un fallo de conmutador o una pérdida de conectividad entre el controlador y los componentes del plano de datos, el método comprendiendo los pasos de:

a) hacer el gráfico (401) de una topología de la red de arquitectura dividida, con enlaces en la red de arquitectura dividida representados como un conjunto de bordes en un gráfico y elementos (317) de red en la red de arquitectura dividida representados como un conjunto de nodos (S1-S10) en los que el tráfico de control se envía desde y hacia el controlador a través de un árbol de enrutamiento de controlador enraizado en el controlador;

b) para cada nodo (S1-S10) del gráfico

generar un árbol de enrutamiento de controlador con cada nodo dicho que sirve como la raíz del árbol

y recorrer (403) el conjunto de nodos (S1-S10) dentro del gráfico para calcular una métrica de protección para cada nodo, en el que la métrica de protección mide la resistencia de la red de arquitectura dividida como un grado de protección de fallo de nodo dentro de la red de arquitectura dividida para una colocación del controlador potencial, el grado de protección de fallo del nodo que determina un subconjunto de nodos protegidos en el conjunto de nodos, o en el que dicha métrica de protección es el número de nodos vecinos protegidos,

donde un nodo protegido puede encapsular el tráfico de control dentro de un mensaje de datos y redirigir dicho tráfico de control a través de un túnel a un nodo intermedio en el gráfico que no está en sentido descendente del nodo protegido, y en el que dicho tráfico de control encapsulado es desencapsulado, y donde el túnel recorre al menos un nodo descendente del nodo protegido; y

seleccionar (405) el elemento (317, 315) de red correspondiente al nodo que sirve como una raíz del árbol con una mejor métrica de protección para ser el controlador (315) para la red de arquitectura dividida.

2.- El método de la reivindicación 1, en el que recorrer el conjunto de nodos para calcular la métrica de protección comprende además el paso de calcular un árbol de enrutamiento para cada nodo en el conjunto de nodos con cada nodo en una raíz de un árbol de enrutamiento correspondiente, teniendo cada árbol de enrutamiento una estructura de árbol con cada nodo que solo tiene un nodo ascendente inmediato en la estructura de árbol.

3.- El método de la reivindicación 2, en el que recorrer el conjunto de nodos para calcular la métrica de protección comprende además los pasos de determinar la ponderación del árbol de enrutamiento basándose en un número de nodos descendentes desprotegidos en el árbol de enrutamiento.

4.- El método de la reivindicación 3, en el que recorrer el conjunto de nodos para calcular la métrica de protección comprende además el paso de sumar todas las ponderaciones de nodo en cada árbol de enrutamiento para obtener la métrica de protección para cada nodo correspondiente en la red de arquitectura dividida.

5.- El método de la reivindicación 4, en el que seleccionar el elemento de red correspondiente al nodo con una mejor métrica de protección para que sea el controlador (315) de la red de arquitectura dividida comprende además el paso de seleccionar el nodo con una ponderación de nodo mínimo para un árbol de enrutamiento correspondiente entre todas las ponderaciones de nodo para todos los árboles de enrutamiento correspondientes al conjunto de nodos en la red de arquitectura dividida.

6.- El método de la reivindicación 1, en el que recorrer el conjunto de nodos para calcular la métrica de protección comprende además el paso de ordenar el conjunto de nodos en orden descendente basándose en un número de enlaces a nodos vecinos para cada nodo.

7.- El método de la reivindicación 6, en el que recorrer el conjunto de nodos para calcular la métrica de protección comprende además el paso de determinar un número de nodos vecinos protegidos con una conexión a otros nodos.

8.- El método de la reivindicación 7, en el que seleccionar el elemento de red correspondiente al nodo con una mejor métrica de protección para que sea el controlador (315) para la red de arquitectura dividida comprende el paso de seleccionar un nodo con el mayor número de nodos vecinos protegidos para que sea el controlador (315).

9.- Una red con una arquitectura dividida donde los componentes del plano de control de la red de arquitectura dividida son ejecutados por un controlador (315) y los componentes del plano de control están separados de los

componentes del plano de datos de la red de arquitectura dividida, en la que la colocación del controlador (315) se selecciona para minimizar la interrupción de la red de arquitectura dividida causada por un fallo de enlace, un fallo de conmutación o una pérdida de conectividad entre el controlador (315) y los componentes del plano de datos, comprendiendo la red:

a) un conjunto de elementos (317) de red interconectados por un conjunto de enlaces de comunicación, cada elemento de red en el conjunto de elementos de red ejecutando un conmutador (317) que está controlado por y en comunicación con el controlador (315); y el controlador ejecutado por uno de los conjuntos de elementos de red, en el que una posición del elemento de red en el conjunto de elementos de red dentro de la red de arquitectura dividida proporciona un número optimizado de enlaces protegidos entre el controlador (315) y cada uno de los elementos (317) de red en el conjunto de elementos de red, el número optimizado correspondiente a una mejor métrica de protección para el elemento de red en el conjunto de elementos de red, en el que una topología de la red de arquitectura dividida es tal que los enlaces en la red de arquitectura dividida se representan como un conjunto de bordes en un gráfico y los elementos (317) de red en la red de arquitectura dividida se representan como un conjunto de nodos (S1-S10), en el que el tráfico de control se envía hacia y desde el controlador a través de un árbol de enrutamiento de controlador enraizado en el controlador;

b) en el que la métrica de protección mide la resistencia de la red de arquitectura dividida como un grado de protección de fallo de nodo dentro de la red de arquitectura dividida para una colocación del controlador potencial, el grado de protección de fallo de nodo determinando un subconjunto de elementos de red protegidos en el conjunto de elementos de red, o en el que la métrica de protección es el número de nodos vecinos protegidos;

donde un elemento de red protegido o nodo protegido puede encapsular el tráfico de control dentro de un mensaje de datos y redirigir el tráfico de control encapsulado a través de un túnel a un nodo intermedio en el gráfico que no está en sentido descendente del elemento de red protegido, y en el que dicho tráfico de control encapsulado es desencapsulado, y donde el túnel recorre al menos un elemento de red descendente del elemento de red protegido.

10.- La red de la reivindicación 9, en la que el conjunto de elementos de red forma un plano de datos de un núcleo de paquete evolucionado (EPC) en una red de evolución a largo plazo (LTE), y el controlador (315) proporciona un plano de control del EPC en la red LTE.

11.- La red de la reivindicación 9, en la que el conjunto de elementos (317) de red forma un conjunto de planos de datos para una pluralidad de tecnologías de red celular, y el controlador proporciona un plano de control para cada una de la pluralidad de tecnologías de red celular.

12.- Un sistema informático para determinar una colocación de un controlador (315) para una red de arquitectura dividida donde los componentes del plano de control de la red de arquitectura dividida son ejecutados por el controlador (315) y los componentes del plano de control están separados de los componentes del plano de datos de la red de arquitectura dividida, en el que el tráfico de control enviado hacia y desde el controlador a través de un árbol de enrutamiento de controlador enraizado en el controlador; en el que la colocación del controlador (315) se selecciona para minimizar la interrupción de la red de arquitectura dividida causada por un fallo de enlace, un fallo de conmutador o una pérdida de conectividad entre el controlador y los componentes del plano de datos, el sistema informático comprendiendo:

a) un procesador (303) configurado para ejecutar un módulo (305) de gráfico de topología y un módulo (307) de colocación del controlador, el módulo (305) de gráfico de topología configurado para hacer el gráfico una topología de la red de arquitectura dividida, con enlaces en la red de arquitectura dividida representados como un conjunto de bordes en un gráfico y elementos (317) de red en la red de arquitectura dividida representados como un conjunto de nodos (S1-S10);

b) el módulo (307) de colocación del controlador configurado para generar para cada nodo (S1-S10) en el gráfico un árbol de enrutamiento de controlador con cada dicho nodo sirviendo como la raíz del árbol

y recorrer el conjunto de nodos dentro del gráfico para calcular una métrica de protección para cada nodo (S1-S10), en el que la métrica de protección mide la resistencia de la red de arquitectura dividida como un grado de protección de fallo de nodo dentro de la red de arquitectura dividida para una colocación del controlador potencial, el grado de protección de fallo de nodo determinando un subconjunto de nodos protegidos en el conjunto de nodos, o en el que dicha métrica de protección es el número de nodos vecinos protegidos;

donde un nodo protegido en el subconjunto de nodos protegidos puede encapsular el tráfico de control dentro de un mensaje de datos y redirigir dicho tráfico de control encapsulado a través de un túnel a un nodo intermedio en el gráfico que no está en sentido descendente del nodo protegido, y en el que dicho tráfico de control encapsulado se desencapsula, y donde el túnel recorre al menos un nodo descendente del nodo protegido,

c) el módulo (307) de colocación del controlador está configurado además para seleccionar el elemento de red correspondiente a un nodo que sirve como una raíz del árbol con la mejor métrica de protección para ser el controlador (315) de la red de arquitectura dividida.

13.- El sistema informático de la reivindicación 12, en el que el módulo (307) de colocación del controlador está configurado además para calcular un árbol de enrutamiento para cada nodo en el conjunto de nodos con cada nodo en una raíz de un árbol de enrutamiento correspondiente, cada dicho árbol de enrutamiento teniendo una estructura de árbol con cada nodo teniendo solo un nodo ascendente inmediato en la estructura de árbol.

14.- El sistema informático de la reivindicación 12, en el que el módulo (307) de colocación del controlador está configurado además para determinar la ponderación del árbol de enrutamiento basándose en un número de nodos descendentes desprotegidos en el árbol de enrutamiento.

15.- El sistema informático de la reivindicación 14, en el que el módulo (307) de colocación del controlador está configurado además para sumar todas las ponderaciones de nodo en cada árbol de enrutamiento para obtener la métrica de protección para cada nodo correspondiente en la red de arquitectura dividida.

16.- El sistema informático de la reivindicación 15, en el que el módulo (307) de colocación del controlador está configurado además para seleccionar el nodo con una ponderación de nodo mínimo para un árbol de enrutamiento correspondiente entre todas las ponderaciones de nodos para todos los árboles de enrutamiento correspondientes al conjunto de nodos en la red de arquitectura dividida.

17.- El sistema informático de la reivindicación 12, en el que el módulo (307) de colocación del controlador está configurado además para ordenar el conjunto de nodos en orden descendente basándose en un número de enlaces a nodos vecinos para cada nodo.

18.- El sistema informático de la reivindicación 17, en el que el módulo (307) de colocación del controlador está configurado además para determinar un número de nodos vecinos protegidos con una conexión a otros nodos.

19.- El sistema informático de la reivindicación 18, en el que el módulo (307) de colocación del controlador está configurado además para seleccionar un nodo con el mayor número de nodos vecinos protegidos para que sea el controlador (315).

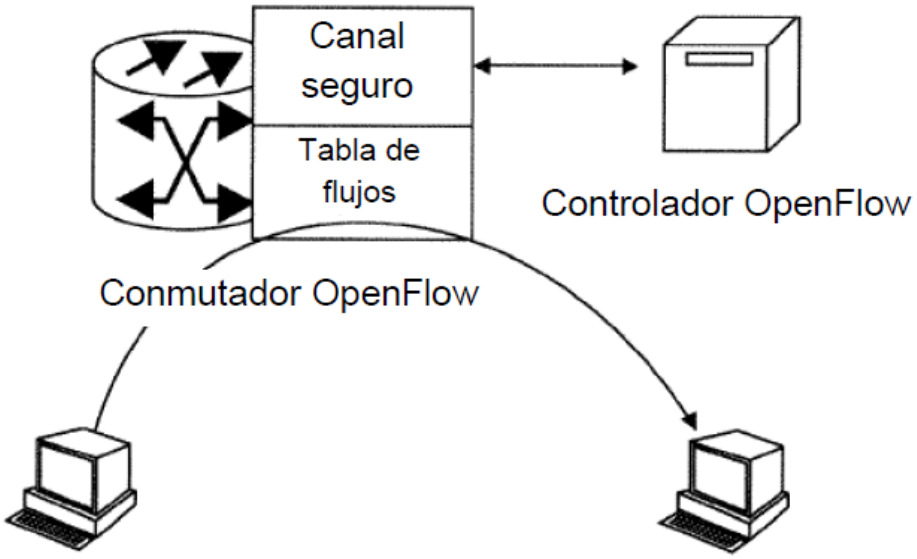
Canal seguro
Tabla de flujos
Controlador OpenFlow
Conmutador OpenFlow

FIG. 1

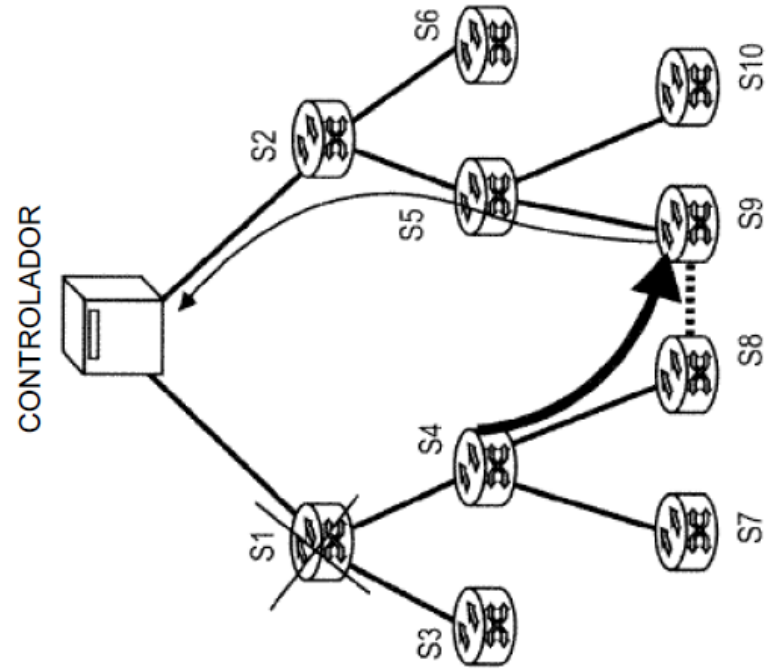
CONTROLADOR
S1
S2
S3
S4
S5
S6
S7
S8
S9
S10

FIG. 2B

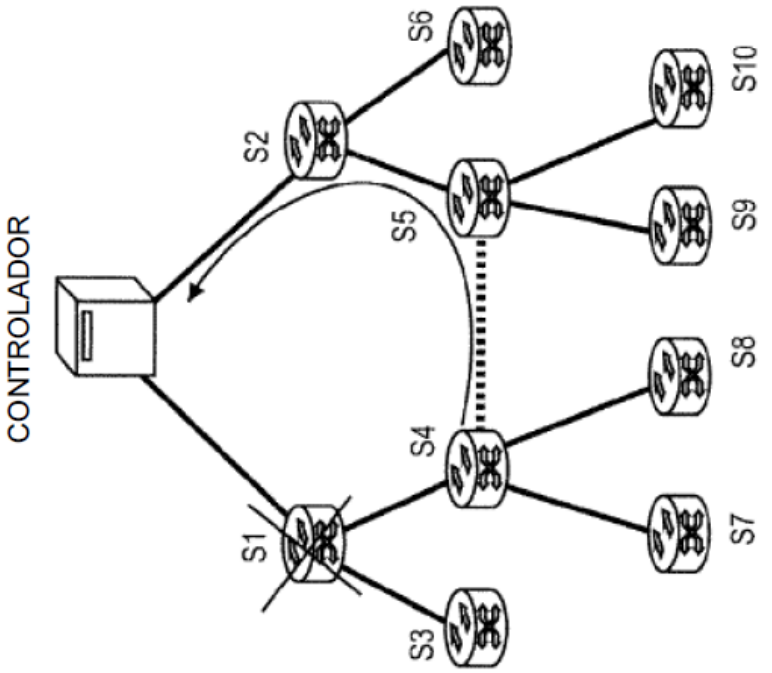
CONTROLADOR
S1
S2
S3
S4
S5
S6
S7
S8
S9
S10

FIG. 2A

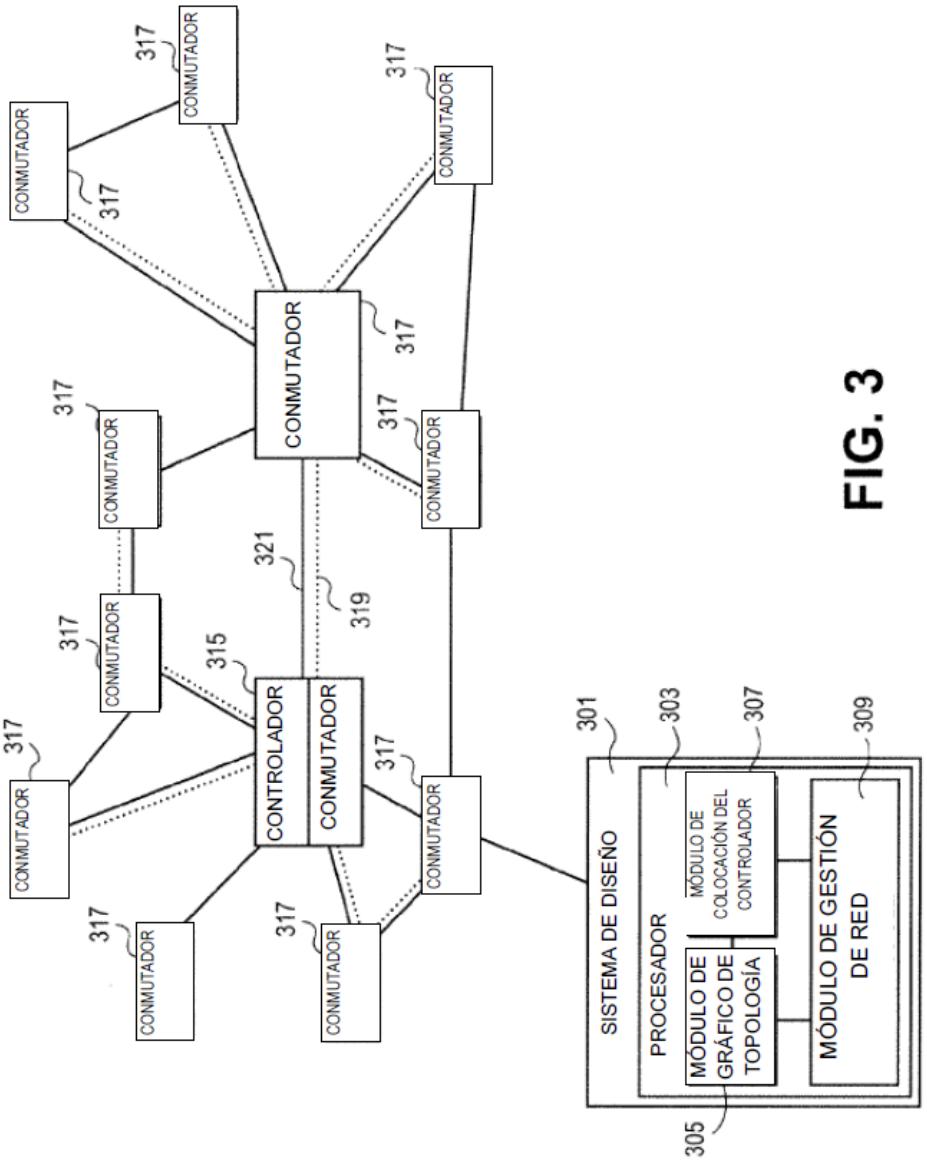
CONMUTADOR
317
CONTROLADOR
CONMUTADOR
315
321
319
301
303
307
309
SISTEMA DE DISEÑO
PROCESADOR
MÓDULO DE GRÁFICO DE TOPOLOGÍA
MÓDULO DE COLOCACIÓN DEL CONTROLADOR
MÓDULO DE GESTIÓN DE RED
305

FIG. 3

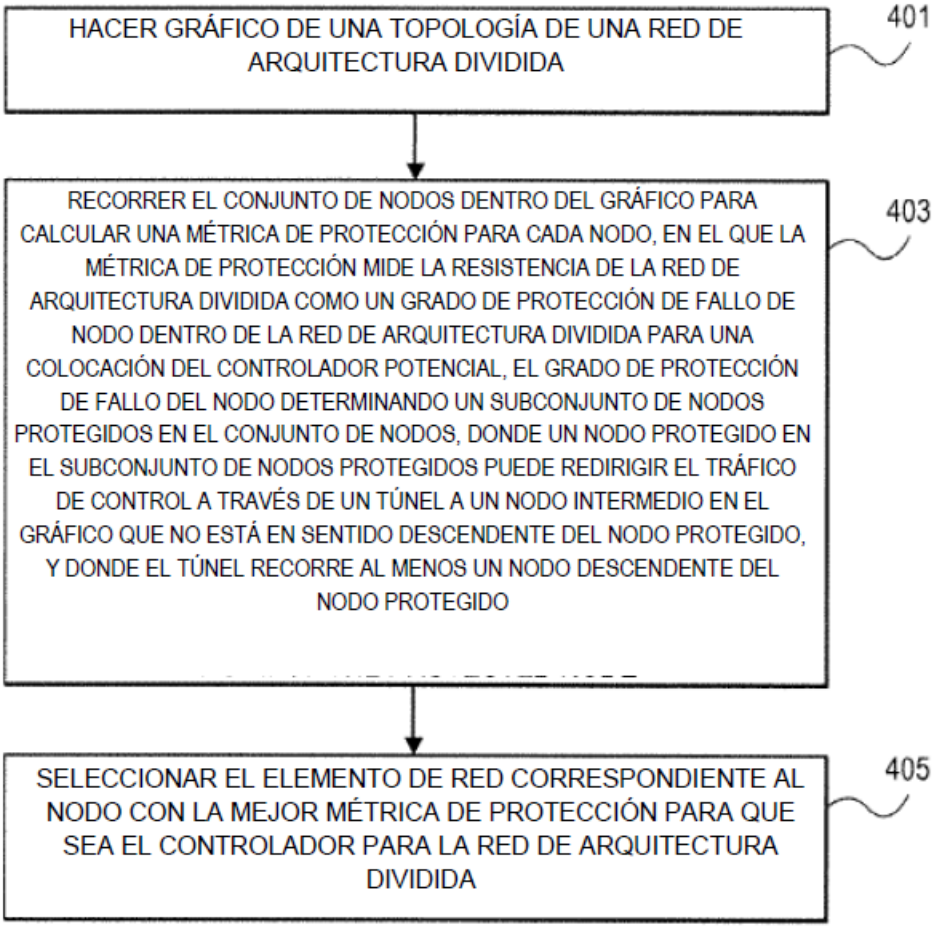
HACER GRÁFICO DE UNA TOPOLOGÍA DE UNA RED DE ARQUITECTURA DIVIDIDA
401
RECORRER EL CONJUNTO DE NODOS DENTRO DEL GRÁFICO PARA CALCULAR UNA MÉTRICA DE PROTECCIÓN PARA CADA NODO, EN EL QUE LA MÉTRICA DE PROTECCIÓN MIDE LA RESISTENCIA DE LA RED DE ARQUITECTURA DIVIDIDA COMO UN GRADO DE PROTECCIÓN DE FALLO DE NODO DENTRO DE LA RED DE ARQUITECTURA DIVIDIDA PARA UNA COLOCACIÓN DEL CONTROLADOR POTENCIAL, EL GRADO DE PROTECCIÓN DE FALLO DEL NODO DETERMINANDO UN SUBCONJUNTO DE NODOS PROTEGIDOS EN EL CONJUNTO DE NODOS, DONDE UN NODO PROTEGIDO EN EL SUBCONJUNTO DE NODOS PROTEGIDOS PUEDE REDIRIGIR EL TRÁFICO DE CONTROL A TRAVÉS DE UN TÚNEL A UN NODO INTERMEDIO EN EL GRÁFICO QUE NO ESTÁ EN SENTIDO DESCENDENTE DEL NODO PROTEGIDO, Y DONDE EL TÚNEL RECORRE AL MENOS UN NODO DESCENDENTE DEL NODO PROTEGIDO
403
SELECCIONAR EL ELEMENTO DE RED CORRESPONDIENTE AL NODO CON LA MEJOR MÉTRICA DE PROTECCIÓN PARA QUE SEA EL CONTROLADOR PARA LA RED DE ARQUITECTURA DIVIDIDA
405

FIG. 4

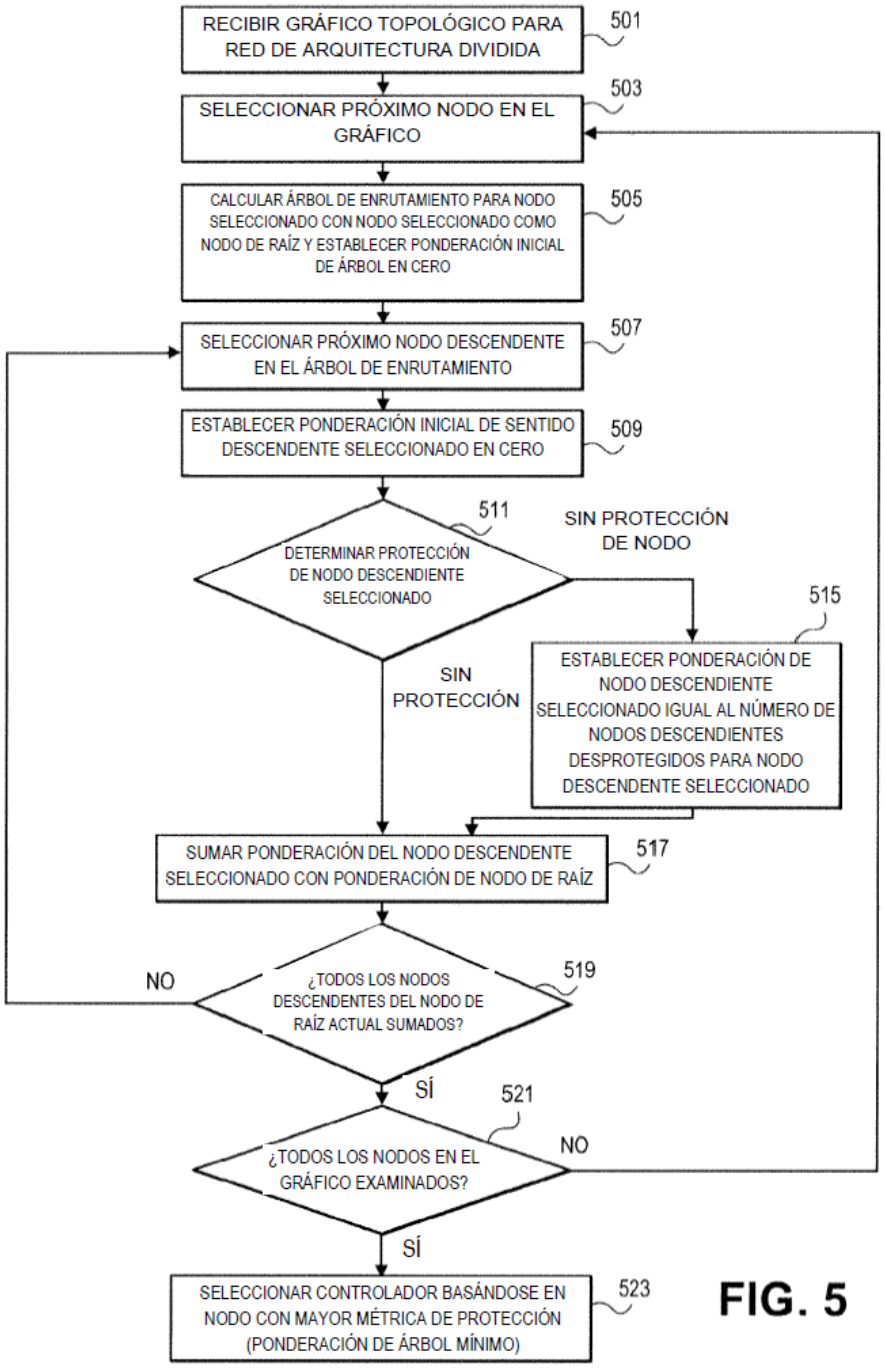
RECIBIR GRÁFICO TOPOLÓGICO PARA RED DE ARQUITECTURA DIVIDIDA
501
SELECCIONAR PRÓXIMO NODO EN EL GRÁFICO
503
CALCULAR ÁRBOL DE ENRUTAMIENTO PARA NODO SELECCIONADO CON NODO SELECCIONADO COMO NODO DE RAÍZ Y ESTABLECER PONDERACIÓN INICIAL DE ÁRBOL EN CERO
505
SELECCIONAR PRÓXIMO NODO DESCENDENTE EN EL ÁRBOL DE ENRUTAMIENTO
507
ESTABLECER PONDERACIÓN INICIAL DE SENTIDO DESCENDENTE SELECCIONADO EN CERO
509
DETERMINAR PROTECCIÓN DE NODO DESCENDIENTE SELECCIONADO
511
SIN PROTECCIÓN DE NODO
SIN PROTECCIÓN
ESTABLECER PONDERACIÓN DE NODO DESCENDIENTE SELECCIONADO IGUAL AL NÚMERO DE NODOS DESCENDIENTES DESPROTEGIDOS PARA NODO DESCENDENTE SELECCIONADO
515
SUMAR PONDERACIÓN DEL NODO DESCENDENTE SELECCIONADO CON PONDERACIÓN DE NODO DE RAÍZ
517
¿TODOS LOS NODOS DESCENDENTES DEL NODO DE RAÍZ ACTUAL SUMADOS?
519
NO
SÍ
¿TODOS LOS NODOS EN EL GRÁFICO EXAMINADOS?
521
NO
SÍ
SELECCIONAR CONTROLADOR BASÁNDOSE EN NODO CON MAYOR MÉTRICA DE PROTECCIÓN (PONDERACIÓN DE ÁRBOL MÍNIMO)
523

FIG. 5

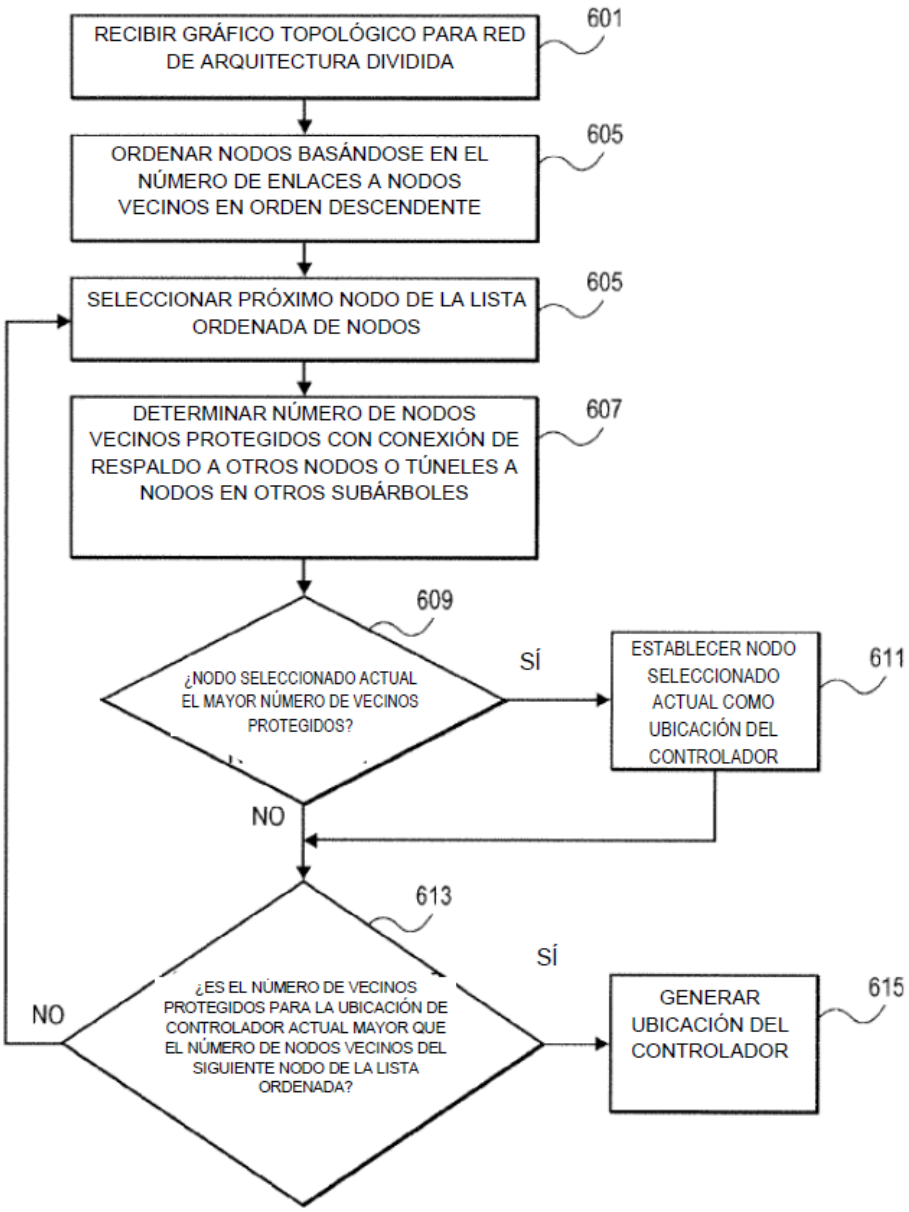
RECIBIR GRÁFICO TOPOLÓGICO PARA RED DE ARQUITECTURA DIVIDIDA
601
ORDENAR NODOS BASÁNDOSE EN EL NÚMERO DE ENLACES A NODOS VECINOS EN ORDEN DESCENDENTE
605
SELECCIONAR PRÓXIMO NODO DE LA LISTA ORDENADA DE NODOS
605
DETERMINAR NÚMERO DE NODOS VECINOS PROTEGIDOS CON CONEXIÓN DE RESPALDO A OTROS NODOS O TÚNELES A NODOS EN OTROS SUBÁRBOLES
607
609
¿NODO SELECCIONADO ACTUAL EL MAYOR NÚMERO DE VECINOS PROTEGIDOS?
SÍ
ESTABLECER NODO SELECCIONADO ACTUAL COMO UBICACIÓN DEL CONTROLADOR
611
NO
613
¿ES EL NÚMERO DE VECINOS PROTEGIDOS PARA LA UBICACIÓN DE CONTROLADOR ACTUAL MAYOR QUE EL NÚMERO DE NODOS VECINOS DEL SIGUIENTE NODO DE LA LISTA ORDENADA?
SÍ
NO
GENERAR UBICACIÓN DEL CONTROLADOR
615

FIG. 6