**Marco conceptual (Nivel 1)**

* **Redes OBS, OCS y redes hibridas.**
* **procesos de decisión de marco continuos.**
* **Enrutamiento y agentamiento de conexiones en redes que soportan reservaciones anticipadamente.**

**DESCRIPCION DE ELEMENTOS CONSTITUTIVOS DEL MODELO AG2**

**DESCRIPCION GENERAL**

El modelo AG2 es una propuesta de modelo arquitectural basado en la tecnología GMPLS y en la infraestructura de computación distribuida[[1]](#footnote-1) (grilla computacional). Siendo el principal objetivo de AG2 la optimización de los recursos de red y adaptar el comportamiento del modelo al tipo y la cantidad de tráfico que por medio de la red fluye.

La descripción física del modelo consta:

1. El punto de grilla:
   1. Puntos de procesamiento.
   2. Puntos de almacenamiento masivo (no es raro encontrarlos en un lugar geográfico distante de su punto de procesamiento).
2. Infraestructura óptica de la red GMPLS.
   1. Enlaces de red.
   2. Enrutadores de borde y de núcleo.

Y la descripción conceptual consta:

1. Datos en bruto.
2. Algoritmo “Caracterizador” el cual interpretara la naturaleza del problema que ejecutara la grilla, para generar un conjunto de características de dicho problema.
3. Algoritmo “Fabrica de Restricciones” el cual será alimentado por el conjunto de datos del algoritmo “Caracterizador”, el cual será el responsable de generar un conjunto restricciones que completaran a las restricciones ya existentes de los protocolos de plano de control de la arquitectura GMPLS.

**DESCRIPCION DETALLADA**

En la descripción conceptual del modelo AG2 se nombraron los elementos constitutivos:

* **Datos en bruto:** Son datos en bruto que una tarea puede tomar como entrada (originalmente residen en sitios de almacenamiento masivo).
* **Algoritmo Caracterizador:**Este algoritmo es el encargadode generar características de ***información estadística***, será alimentado periódicamente por el comportamiento de la red y del protocolo usado en el plano de control (véase sección Z), generando así características cuantificables y medibles de comportamiento de la red y de los flujos que pasan a través de ella.
* **Algoritmo Fábrica de Restricciones:** Este algoritmo recibe como entrada los parámetros generados por el “algoritmo caracterizador”, los procesa con ayuda del protocolo CSPF y genera los mejores LSP que se ajusten al requerimiento.
* También debe optimizar el uso de los recursos de la red, la forma en la que se propone hacerlo es reduciendo la gestión que se hace sobre los caminos ópticos (LSPs). La reducción de la gestión consiste en mantener los que sirvan como paso común para más de un flujo o un solo flujo de tráfico y eliminar los sub-utilizados. Al recolectar esta información de tal forma que sea agrupable en franjas de tiempo para de dicha forma poder crear una gestión de LSPs por adelantado y no generar gasto sobre la red haciendo la gestión en forma dinámica.

**Sección Z. MENSAJES DEL PROTOCOLO RSVP-TE (DEDUCCION DE INFORMACION ESTADISTICA)**

En todos los mensajes descritos a continuación hacemos uso de la separación de planos de la arquitectura GMPLS (plano de datos y plano de control), estando los mensajes de señalización del plano de control encapsulados en el datagrama IP, así:



Figura 1 Estructura y encapsulamiento de mensajes RSVP.

Además, también se muestra superficialmente que cada mensaje tiene en encabezado común y unas estructuras y sub-estructuras de objetos transportadores de información particular.

**DEDUCCION DE INFORMACION ESTADISTICA**

1. **¿Tiempo de generación del LSP[[2]](#footnote-2)?**

El enrutador de ingreso (que es un enrutador de borde) es el encargado de generar un mensaje de camino (Path message) a su siguiente salto (enrutador de núcleo) en el camino del LSP. Este determina el camino ya sea viendo su ruta explicita o computando el siguiente salto hacia su destino, este mensaje de solicitud lleva el identificador de la sesión padre (sesión object) y el identificador del LSP (sessión template), además de los siguientes parámetros que describen al LSP (Label Request, Sender-TSpec y Explicit Route objects) además verificando salto a salto que los recursos si puedan soportar el LSP. Cuando el mensaje de camino alcanza el enrutador de la cadena de bajada (“DownStream router” que es un enrutador de borde), este ultimo genera un mensaje de aceptación (Resv message) para provisionar las etiquetas que deben ser usadas para identificar el trafico (Objeto etiqueta “Label object”) y para confirmar la reservación de los recursos (“FlowSpec object” objetos que especifican el tipo de reservación y la información de la reservación). Al llegar dicho mensaje de aceptación al enrutador de cadena de subida (“upstream router” o más conocido como enrutador de ingreso) se da por establecido el LSP. En conclusión, podemos calcular el tiempo que tarde el establecimiento del LSP si tomamos el tiempo en el enrutador de ingreso cuando es enviado el mensaje de camino “Path message” y le restamos el tiempo de llegada a este mismo enrutador del mensaje de aceptación “Resv message”, mas formalmente así:

tpm : Tiempo de cuando se genero el mensaje de camino en el enrutador de ingreso.

tcm : Tiempo de cuando arribo el mensaje de aceptación al enrutador de ingreso.

TeLSP : Tiempo de establecimiento del LSP.

Siendo TeLSP = tcm -tpm.

El tiempo como parámetro obtenido de un LSR de borde es obtenido con una solicitud de timestamp, dicha solicitud es parte del protocolo ICMP, el cual le brinda la hora del reloj.

1. **¿Tiempo de duración de LSP?:**

Enfocándonos en el enrutador de ingreso (“Upstream router”) el cual es el que genera el LSP con el mensaje Path, podemos calcular el tiempo de duración del LSP si sabemos el momento de arribo del mensaje Resv hasta el momento de generación del mensaje de liberación de cadena de bajada (“DownStream Release” teniendo en cuenta que los protocolos RSVP y RSVP-TE no tienen respuesta a este mensaje), ya que este mensaje viaja salto a salto por la cadena de bajada hasta el enrutador de egreso.

Si bien mencionamos en líneas anteriores que el mensaje de liberación de cadena de bajada no tiene respuesta, sino que simplemente va liberando los recursos atados al LSP. Por otro lado, la arquitectura GMPLS si tiene en cuenta un mensaje, no propiamente dicho de respuesta, si no un mensaje que permita a un enrutador de egreso (“Downstream router”) poder terminar un LSP, dicho mensaje utiliza el mensaje PathErr (Mensaje generado cuando el mensaje Path viaja por la cadena de bajada y no puede cumplir los parámetros de establecimiento del LSP, se genera en la cadena de subida el PathErr) pero adicionándole a dicho mensaje un bandera de “Remoción del estado del camino” (‘‘Path state removed’’). Siendo mas precisos así se calcularía para el caso cuando el enrutador de ingreso es el mismo que desea eliminar el LSP:

trm : Tiempo del momento cuando arribo el mensaje Resv(Mensaje de confirmación de generación del LSP).

Tdsrm: Tiempo de cuando se genero el mensaje de liberación de recursos del LSP.

TvLSP : Tiempo de vida del LSP.

Siendo TvLSP = Tdsrm - trm.

Ahora contemplamos la opción para cuando el enrutador de cadena de bajada es el que quiere terminar el LSP.

trm : Tiempo del momento cuando arribo el mensaje Resv(Mensaje de confirmación de generación del LSP).

TphPSR: Tiempo de arribo del mensaje PathErr con bandera de “Remoción del estado del camino” (‘‘Path state removed’’).

TvLSP : Tiempo de vida del LSP.

Siendo TvLSP = TphPSR - trm.

1. **¿De los LSPs generados cuantos presentaron fallos?**

Una vez un LSP es establecido, hay mensajes Path y Resv de refresco periódicos los cuales son los encargados de dar certeza de la existencia del estado del LSP. Ya que los errores pueden ser reportados con dos tipos de mensajes, el Mensaje de Error de Cadena de Bajada (Downstream error message) y el Mensaje de Error de cadena de subida (Upstream error message), se sabria que un LSP durante su tiempo de vida presento fallas si se estableció y luego se recibió un mensaje de error, por ejemplo, de cadena de subida en el caso del enrutador de ingreso. El mensaje de cadena de subida es el mensaje el cual se usa para reportar errores en LSP ya establecidos (LMP es usado para ayudar a aislar la falla, es decir, determinar donde en el camino sucedió la falla) pero la propagación de la existencia del error se debe hacer por medio del mensaje de error en el plano de control.

Ambos tipos de mensajes de error, el de cadena de subida y el de cadena de bajada llevan consigo información para ayudar a determinar y localizar donde sucedió. Un código clasifica el tipo de error (de admisión, de enrutamiento o de política), al igual también contienen la dirección IP del LSR que reporta el error y pueden usar TLV para indicar el enlace especifico en donde el problema sucedió. Así que para saber cuántos LSP presentaron fallas, necesitamos saber cual LSP fallo y cuantas veces, determinándolo así:

Se sabe que LSP fallo porque sabemos que cada LSP tiene el identificador de la sesión padre (session object) y el identificador del LSP (sender- template). Y sabemos cuántas veces fallo y el tipo de fallo por que el enrutador de ingreso recibe mensajes de error ya mencionados.

1. **¿Cuántos se recuperaron? y ¿cuántos no se recuperaron?**

Sabemos que para el establecimiento de un LSP se genera un Path msg y lo confirma un Resv msg, una vez establecido, hay un tiempo de refresco (en casos hasta de 10 minutos que se encuentra en el objeto “TIME\_VALUES Object”), de generarse algún fallo antes de querer eliminar un LSP (eliminar el servicio “mensaje de liberación”), ha de producirse un mensaje de error (PathErr o ResvErr con la información en el objeto), según sea el caso se clasifica el error y el enrutador de ingreso decide si se no se desea mantener más tiempo el servicio (mantener el LSP) generando un mensaje de liberación o si por el contrario el plano de control logro solucionar el inconveniente, (vale la pena recordar que en el mensaje de error tenemos información del tipo de error y en donde sucedió “Error\_Spec\_Object”[[3]](#footnote-3)), así que se puede determinar cuántos LSP se recuperaron a un fallo y cuantos se perdieron leyendo la información del mensaje de error (PathErr) que arriba al enrutador de ingreso. No se que tan necesario seria examinar la opción q después de recibir un mensaje de error, los LSPs q se eliminaron cuantificarlos teniendo en cuenta con el mensaje de liberación del servicio (Release msg).

1. **¿Cuantos LSP se generaron (Origen-Destino)?:Mirar SCOPE Object pag 12**

En el plano de control podemos identificar los nodos origen-destino (Enrutador de ingreso y de egreso) ya que estos mensajes están encapsulados dentro de un datagrama IP (un datagrama contiene la dirección IP de origen y destino), si bien lo anterior resuelve el problema del origen y del destino de un LSP (de enrutador de ingreso a enrutador de egreso), no sabe que trafico es el que viaja dentro de ese LSP, ya que desde un mismo origen de LSP pude venir varios flujos, este problema se resuelve dentro de la estructura del mismo flujo. Una vez un flujo se inicia, y el paquete IP entra en el núcleo GMPLS se le incrusta una etiqueta (una longitud de onda), si bien el enrutamiento del paquete se hace basándose en la etiqueta recién incrustada, el datagrama IP con sus puntos de origen y destino de red siguen estando presentes, lo cual nos brinda información de origen a destino. Al leer dicha información en cada enrutador para saber sus redes tanto de origen y destino en principio se pensaría que se va a perder en algo la eficiencia que se gana con el enrutamiento basado en etiquetas, resulta que dicha información solo se leería para saber el destino del trafico mas no se operaria sobre dichas direcciones IP para algún proceso relacionado con el enrutamiento.

1. **¿Cuanto trafico paso por el LSP?:**
2. **¿Que ancho de banda tiene el LSP?:**

En el mensaje de solicitud de establecimiento del LSP (Path message) además de llevar consigo el identificador de sesión (sesión object) y el identificador del LSP (Sender\_template) tambien lleva los parámetros de que describen los requerimientos del LSP (Label Request, Sender-TSpec, y Explicit Route objects). De los anteriores el que describe los parámetros del tráfico planeado es el Sender-TSpec, este objeto es un descriptor de tráfico que describe los requerimientos de ancho de banda o puede detallar otros parámetros de calidad de servicio (QoS). A su vez es confirmado con el objeto FlowSpec que representa el compromiso de la reservación de recursos que van a soportar el flujo de tráfico.

1. **¿Por cuales nodos paso el LSP?**

La lista de nodos por los cuales pasa el LSP se puede determinar a lo largo de la vida del mismo, ya sea en su establecimiento como en su mantenimiento, por que el objeto recRoute hace parte de los mensajes Path y Resv, teniendo este objeto una lista de direcciones que el mensaje Path ha transmitido[[4]](#footnote-4).

**ACLARACION OBJETO SCOPE**: Este objeto lleva una lista explícita (detallada) de los nodos a los cuales la información en el mensaje es enviada (Diferente a objeto recRoute).

**TABLA RESUMEN PARAMETROS DE INFORMACION ESTADISTICA**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **PARAMETRO** | **TIPO DE DATO** | **UNIDAD** | **OBJETO** | **MENSAJE** | **PROCEDIMIENTO** | **DOMINIO** |
| Tiempo de generación del LSP | Numérico | ms | Ninguno | Path message y Resv message | tpm: Tiempo en el cual se genero el mensaje de camino en el enrutador de ingreso.  tcm: Tiempo en el cual arribo el mensaje de aceptación al enrutador de ingreso.  TeLSP: Tiempo de establecimiento del LSP.  Siendo TeLSP = tcm -tpm. | 8 bits |
|  |  |  |  |  |  |  |
| Tiempo de duración de LSP | Numérico | m | Ninguno | Resv message, DownStream release message y PathErr message. | trm: Tiempo del momento cuando arribo el mensaje Resv(Mensaje de confirmación de generación del LSP).  Tdsrm: Tiempo de cuando se genero el mensaje de liberación de recursos del LSP.  TvLSP: Tiempo de vida del LSP.  Siendo TvLSP = Tdsrm - trm.  Ahora contemplamos la opción para cuando el enrutador de cadena de bajada es el que quiere terminar el LSP.  trm : Tiempo del momento cuando arribo el mensaje Resv(Mensaje de confirmación de generación del LSP).  TphPSR: Tiempo de arribo del mensaje PathErr con bandera de “Remoción del estado del camino” (‘‘Path state removed’’).  TvLSP : Tiempo de vida del LSP.  Siendo TvLSP = TphPSR - trm. | 10 bits |
| Numero de LSPs que fallaron. | Numérico | Ninguno | Session object(variable Tunnel ID) y Sender template object(variable LSP ID) | Path message(Sender template object), todos los mensajes contienen el objeto Session. | Se sabe que LSP fallo porque sabemos que cada LSP tiene el identificador de la sesión padre (session object “variable Tunnel ID”) y el identificador del LSP (sender- template object). | Enteros positivos |
| Numero de LSPs recuperados y perdidos. | Numérico | Ninguno | Error\_Spec Object(información del error) y sesión object (variable unnel ID) | PathErr message, ResvErr message y DownStream Release message | Leyendo la información del mensaje de error (PathErr) que arriba al enrutador de ingreso se sabe que tunnel fallo (No que LSP). Y luego cuántos servicios de tunnel se deciden eliminar con el mensaje Release (PathTear). | Enteros posivios. |
| Numero de LSP que se generaron entre el mismo origen y destino | Numérico | Ninguno | Sender template object(LSP ID) | Path message | Se lee el datagrama IP en sus direcciones origen-destino y adicionalmente de generarse más de un LSP entre el mismo origen-destino debe leerse del mensaje Path el objeto Sender template que contiene el LSP ID. | Enteros posivios. |
| Ancho de banda de un LSP. | Numérico. | mbps | Sender-TSpec | Path message | Una lambda en CWDM puede transportar hasta 2.5 Gbps y en DWDM hasta 10 o inclusive 40 Gbps. | 16 bits |
| Nodos recorridos por el LSP | Numérico (Lista). | Ninguno | recRoute object | Path message y Resv message. | Se lee la lista de nodos que están almacenados en el objeto recRoute. | Direcciones IP (32 bits). |

**Tabla de unidades:**

|  |  |
| --- | --- |
| **UNIDAD** | **DESCARIPCION** |
| ms | Mili-segundos |
| m | minutos |
| mbps | Mega-bits por segundo |
| Lista | Sucesión de números separados por un símbolo o testimonio. |

**Arregla intro, tabla de unidades, mirar lo de amarillito y arreglar unidades(entero positivo). Decir q es objeto y mensaje.**

DESCRIPCIÓN DE LAS INTERFACES DE LA SALIDA

## Descripción restricciones para la computación de camino

El mecanismo el cual genere solicitudes de establecimiento y finalización de los LSP’s del modelo AG2 debe poder generar valores que cumplan con el estándar actual de las restricciones del proceso de computación de camino (PCC[[5]](#footnote-5)) basado en restricciones de la arquitectura GMPLS, para lo cual se hace necesario hacer una introducción conceptual de los diferentes tipos de restricciones, las cuales se clasifican en:

* Restricciones de tipo-enlace.
* Restricciones de tipo-camino.
* Restricciones basadas en inclusiones.
* Restricciones basadas en exclusiones.

### Atributos del enlace en GMPLS

Para cumplir los parámetros de los algoritmos de computación de camino basado en restricciones de la arquitectura GMLPS, el algoritmo fábrica de restricciones, debe generar restricciones basadas en los diferentes atributos del enlace, los cuales están estandarizados en GMPLS[[6]](#footnote-6).

* Métrica de ingeniera de tráfico.
* Grupo de administración.
* Tipo de protección de enlace.
* Grupo de enlaces riesgo compartido.(*Shared Risk Link Group, SRLG*).
* Descriptor de la capacidad de computación de la interface. (Interface Switching Capability, ISC)

#### Métrica de ingeniera de tráfico

Es usado como un costo de la arista direccionada que representa el enlace TE sobre el grafo de la red, este valor numérico es establecido y publicado por el controlador del conmutador, a su vez cada controlador de red introduce este costo a su grafo de red. Esta métrica es usada para establecer los posibles caminos de menor costo.

#### Grupo de administración

Este atributo es un número de 32 bits, que puede ser publicado por cada enlace. Cada bit representa una calidad conocida del enlace, así el proceso de computación de camino puede ser restringido para evitar o forzar el uso de ciertos enlaces por su calidad, dependiendo del tipo de servicio requerido[[7]](#footnote-7).

#### Tipo de protección de enlace.

Este atributo representa la capacidad que tiene un enlace para brindar y recibir ciertas protecciones. Este atributo es usado como restricción al considerar enlaces que garanticen un aceptable nivel de protección del tipo de servicio requerido. Todas las actuales capacidades de protección son presentadas en la siguiente tabla.

Tabla de los tipos de capacidades de protección de los enlaces GMPLS[[8]](#footnote-8)

|  |  |  |
| --- | --- | --- |
| Atributo | Capacidad | Descripción |
| 0x01 | Trafico Extra | Este enlace es establecido para proteger uno más enlaces, otros servicios pueden usar el enlace pero serán interrumpidos si falla alguno de los enlaces que están protegiendo. |
| 0x02 | Sin protección | Este enlace no cuenta con ninguna protección, si el enlace falla el servicio será suspendido. |
| 0x03 | Compartido | Existe uno o más enlaces disyuntos de “trafico extra” que protegen este enlace, además otros enlaces “compartidos” son a su vez protegidos por el mismo enlace de “trafico extra”. |
| 0x04 | Dedicado 1:1 | Existe un enlace disyunto de “extra trafico” para proteger este enlace y la protección no es compartida con otros enlaces. |
| 0x05 | Dedicado 1+1 | Existe un enlace disyunto de “extra trafico” para proteger este enlace y la protección no es compartida con otros enlaces, además el enlace de protección no es publicado por los protocolos de enrutamiento, otros servicios no pueden usarlo. |
| 0x06 | Mejorado | Esquemas de protección avanzadas protegen este enlaces tales como BLSR/MS-SPRING |

#### Grupo de enlaces riesgo compartido

Este atributo identifica todos los grupos de riesgo compartido a los cuales un enlace pertenece, este riesgo se refiere a la posibilidad de que estos enlaces puedan ser afectados cuando un recurso común de red falle. Cada enlace puede pertenecer a múltiples *SRLGs,* así cada atributo de enlace SRLG puede incluir más de un identificador *SRLG*. Este identificador es un número único de 32 bits.

Este atributo es bastante útil para la recuperación del LSP. Es siempre deseable para el servicio de recuperación de LSP, usar un camino los mas disyunto posible del servicio original de establecido del LSP, así una falla en la red no afecte ambos LSP’s, es fácil concluir que debemos restringir el proceso de computación de camino para asegurar que los LSP no compartan SRLG’s.

#### Descriptor de la capacidad de computación de la interface (ISC)

A pesar que el *ISC* es catalogado como un atributo del enlace, es realmente la descripción el tipo de interface de datos, este atributo contiene la siguiente información:

* Tipo de capacidad de la interface de computación.
* Tipo de codificación de datos.
* Máximo ancho de banda disponible para la reservación del LSP de cada nivel de prioridad.

Para la capacidad de conmutación de paquetes el ISC provee información adicional:

* Ancho de banda mínimo para el LSP.
* Unidad de transferencia máxima de la interface (MTU).

Para la capacidad de computación TDM el ISC provee información adicional:

* Ancho de banda mínimo para el LSP.
* Indicador si el modelo Estándar oArbitrariode *SONET/SDH* es soportado.

Los tipos de capacidades de la interface de computación en GMPLS se describen en la siguiente tabla.

Valores de los tipos de capacidades de conmutación de la interface de datos[[9]](#footnote-9)

|  |  |  |
| --- | --- | --- |
| Identificador | Tipo de capacidad de computación | Descripción |
| 1 | PCS-1 | Capacidad de conmutación de paquetes nivel 1 |
| 2 | PCS-2 | Capacidad de conmutación de paquetes nivel 2 |
| 3 | PCS-3 | Capacidad de conmutación de paquetes nivel 3 |
| 4 | PCS-4 | Capacidad de conmutación de paquetes nivel 4 |
| 51 | L2SC | Capacidad de conmutación de la capa 2 |
| 100 | TDM | Capacidad de conmutación por división de tiempo |
| 150 | LSC | Capacidad de conmutación de lambda |
| 200 | FSC | Capacidad de conmutación de fibra. |

El tipo de codificación de datos esta siempre relacionado con la capacidad de conmutación, este provee información sobre la codificación soportada en la interface, esto hace referencia al formato de presentación para el medio de transporte. La tabla mostrada a continuación describe todos tipos de codificación soportadas por la actual arquitectura GMPLS.

Valores para los tipos de codificación de datos[[10]](#footnote-10)

|  |  |
| --- | --- |
| Identificación | Tipo de codificación de datos |
| 1 | Packet |
| 2 | Ethernet |
| 3 | ANSI/ETSI PDH |
| 4 | Reservado |
| 5 | SDH ITU-T G.707/SONET ANSI T1.105 |
| 6 | Reservado |
| 7 | Digital wrapper |
| 8 | Lambda (photonic) |
| 9 | Fiber |
| 10 | Reservado |
| 11 | Fiber channel |

### 

El máximo ancho de banda disponible para la reservación del LSP de cada nivel de prioridad, se refiere al valor del ancho de banda que está disponible en un enlace para un nuevo servicio de cierto nivel de prioridad, así el PCC puede considerar solamente aquellos que tengan el suficiente ancho de banda.

### Atributos de camino

Los atributos de camino son características del camino o de un segmento del camino que puede influenciar la decisión de aceptación o rechazo en el proceso de selección de camino. Usualmente todos los enlaces que componen el camino contribuyen a generar el valor para el atributo, por lo tanto este puede ser calculado de acuerdo a siguiente fórmula[[11]](#footnote-11):



Donde: “Latr(a)” son los atributos del enlace usados por el camino “p” y representado en el grafo por el arco “a”. A continuación se mencionan algunos atributos de camino.

* Longitud total del camino.
* Total de retraso de extremo a extremo.
* Total de conversiones óptico-eléctrico-óptico.
* La unión de todos los SRLG que constituyen el camino.

### Restricciones del proceso de computación de camino.

#### Exclusiones

Las exclusiones globales son fácilmente implementadas al eliminar del grafo las aristas y los vértices que representan los nodos y vértices que deben ser excluidos, y luego ejecutar el correspondiente algoritmo de computación de camino sobre el grafo modificado.

El problema es bastante complejo cuando se requiere dos o más caminos disyuntos, cada uno con su propio conjunto privado de exclusiones, porque no se pude resolver con los algoritmos tradicionales debido a que estos asumen cierto intercambio de información de segmentos entre los caminos. Este inconveniente se puede resolver de la misma forma en que se resuelve las restricciones de tipo-camino, pero el costo computacional es elevado.

#### Restricciones de tipo-enlace

En este tipo de restricciones se tiene una función la cual toma uno o más atributos del un enlace dado y retorna solo dos posible valores, VERDADERO si el enlace cumple con la condiciones para que se pueda establecer el camino sobre este, si no cumple esta función retorna FALSO, lo que significa que este enlace será eliminado del grafo que representa la red. De esta forma es fácil notar que estas restricciones pueden ser manejadas en una forma similar a las exclusiones globales.

#### Inclusiones

El problema de las inclusiones globales (restringir al proceso de computación de camino para que escoja cierto enlaces o nodos) puede ser formulado como una solicitud para computar uno o más caminos disyuntos, donde cada uno de ellos debe cruzar una lista ordena de nodos. Por lo tanto este problema puede ser resuelto al modificar alguno de los algoritmos de selección de camino sin restricciones.

El algoritmo modificado debe tomar en consideración los siguientes pasos.

1. “Asegurar que en la lista de inclusiones los nodos y enlaces no se repitan.
2. Ejecutar un algoritmo de selección de camino más corto, para determinar el camino desde el origen a la primera inclusión.
3. Si el destino del anterior segmento es también el destino de todo el camino del algoritmo se termina y retorna el camino resultante, diferenciando los diferentes segmentos que generaron las inclusiones.
4. Si no, tomar los vértices elegidos por el anterior segmento y adicionarlos en la lista de exclusiones (esto se hace para evitar *ciclos* ), y ejecutar un algoritmo de selección de camino más corto que puede tener en consideración la lista de exclusiones
5. Si la computación de camino del paso 4 es exitosa, regresar al paso 3.
6. Si no, recursivamente re-computar los caminos para todos los segmentos previos comenzando por el exitoso mas reciente.
7. Si el paso 6 es exitoso ir al paso 3.
8. Si no, terminar el algoritmo, y se concluye que el camino con esas inclusiones no existe.”[[12]](#footnote-12)

#### Restricciones de tipo-camino

Las restricciones de camino no cumplen con el esquema de eliminación de vértices y aristas del grafo de la red a priori del proceso de computación de camino, debido a que los parámetros necesarios para evaluar la condición de la restricción solo son conocidos luego de la tener todo el camino computado.

Las restricciones de camino son muy importantes para el proceso de cómputo de caminos sobre redes ópticas, ya que los caminos pueden atravesar una o más redes transparentes. Debido a que la calidad de la señal óptica en dichas redes puede cambiar dramáticamente entre puntos de regeneración.

Un forma de reducir el impacto en la calidad de las redes ópticas es adicionar una restricción de camino que permita producir un deseado camino, el cual no atraviese un número máximo de redes transparentes o un camino en el cual la longitud de cada segmento entre puntos de regeneración siempre sea menor a cierto valor, para así garantizar cierta calidad de servicio.

## Descripción de las interfaces de salida del modelo AG2

\* Complejidad computacional de la función GMPLS

\*\*El enlace ID; identificador de enlace TE, el cual es único para cada enrutador, así se puede diferenciar cada enlace lambda con una combinación de la dirección IP de la interface y el enlace ID.

# 

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Clasificación** | **Definición/Atributo GMPLS** | **Unidad** | **Tipo de dato** | **Complejidad\*** | **Tipo de dato AG2** | **Capacidad** |
| Restricciones de los atributos de enlace | Métrica de ingeniera de tráfico | Métrica | Entero positivo | Baja | Numérico | 16 bits |
| Grupo de administración | [sin unidad] | Entero positivo | Baja | Numérico | 32 bits |
| Tipo de protección de enlace | Atributo de protección de enlace | Entero positivo | Baja | Numérico | 4-bits |
| Grupo de enlaces riesgo compartido | Atributo SRLG | Lista de identificadores de SRLG's | Baja | lista de valores numéricos | lista finita de elementos de 32 bits |
| ISC-Tipo de conmutación | Identificador de capacidad de conmutación | Entero positivo | Baja | Numérico | 8-bits |
| ISC-Codificación | Identificador de tipo de codificación | Entero positivo | Baja | Numérico | 4-bits |
| ISC-Máximo ancho de banda disponible para el LSP. | bytes/segundo | Entero positivo | Baja | Numérico | 64-bits |
| ISC-Paquete - Mínimo ancho de banda para el LSP. | bytes/segundo | Entero positivo | Baja | Numérico | 64-bits |
| ISC-Paquete - Unidad de transferencia máxima de la interface | bytes | Entero positivo | Baja | Numérico | 16-bits |
| ISC - TDM- Mínimo ancho de banda para el LSP | bytes/segundo | Entero positivo | Baja | Numérico | 64-bits |
| ISC - TDM- Indicador si el modelo Estándar o Arbitrario de SONET/SDH es soportado | [sin unidad] | Entero positivo | Baja | Numérico | 2-bits |
| Restricciones de los atributos del camino | Numero de salto | [sin unidad] | Entero positivo | Media | Numérico | 8-bits |
| Número de redes transparentes | [sin unidad] | Entero positivo | Media | Numérico | 8-bits |
| Distancia máxima entre puntos de regeneración | Metros | Entero positivo | Media | Numérico | 16-bits |
| Retraso máximo de extremo a extremo | Milisegundos | Entero positivo | Media | Numérico | 32-bits |
| Trafico promedio sobre todo el camino. | bytes/segundo | Entero positivo(bytes) | Media | Numérico | 64-bits |
| Total de conversiones óptico-eléctrico-óptico | [sin unidad] | Entero positivo | Media | Numérico | 8-bits |
| La unión de todos los SRLG que constituyen el camino | ?? | ?? | ?? |  |  |
| Inclusiones | Enlaces | Identificador | Dirección IP-4/6 -- Enlace ID \*\* | Alta | IP4/IP6 -Numérico | Enlace ID: 16 bits |
| Enrutadores | Identificador | Dirección IP-4/6 | Alta | IP4/IP6 |  |
| Exclusiones | Enlaces | Identificador | Dirección IP-4/6 -- Enlace ID \*\* | Baja | IP4/IP6 -Numérico | Enlace ID: 16 bits |
| Enrutadores | Identificador | Dirección IP-4/6 | Media | IP4/IP6 |  |

**DEFINICIONES DEL MODELO DINÁMICO AG2 (Nivel 1)**

**MARCO TEORICO (N2)**

**TAXONOMIA DE ALGORITMOS DE AGENDAMIENTO EN GRILLAS (N3)**

La taxonomía de los algoritmos de agendamiento en las grillas se realiza tomando como punto de referencia la característica hacia la cual el algoritmos en particular se enfoca. De esta forma teniendo diversos algoritmos centrados en una gran diversidad de focos. Dicha diversidad nace entre otos factores por la gran heterogeneidad en la composición de las grillas. En la siguiente figura se ve los puntos de referencia en los cuales se centras los distintos algoritmos y más adelante se explica sucintamente como es su funcionamiento.

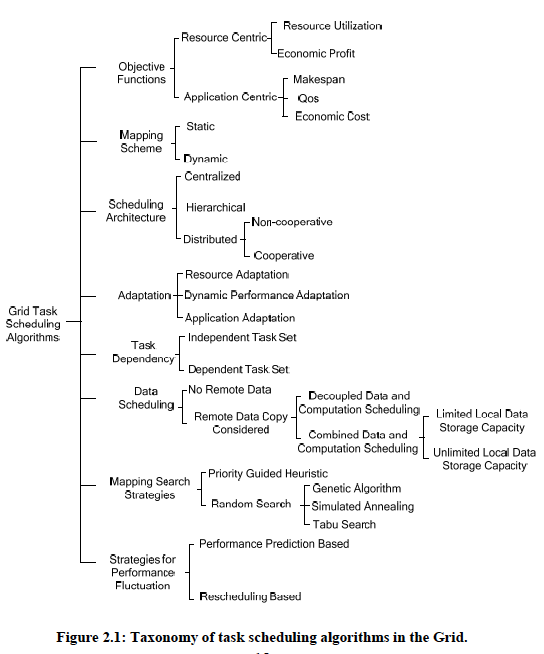
****

Fig. 1 Taxonomía de Algoritmos de Agendamiento de Tareas en la grilla.

**Funciones objetivo:**

En las grillas computacionales se pueden ver en dos grandes partes, en los consumidores de recursos y los proveedores de recursos. Los últimos centran su atención en como los recursos son utilizados y que rédito reciben y los primeros en cómo sus aplicativos se desempeñan. Las motivaciones de ambos para unirse a la grilla se representan por medio de funciones objetivo, siendo clasificadas así: centrados en la aplicación o centradas en los recursos.

Los algoritmos de agentamiento que adoptan función objetivo centrados en la aplicación, tienen como propósito optimizar el funcionamiento de cada trabajo. (Makespan) o tiempo de propagación, es el tiempo gastado desde el inicio de la primera tarea del trabajo a él final de la última tarea del trabajo, el cual (makespan) es usado frecuentemente como una función objetivo en muchos algoritmos de agentamiento.

En el mercado de productos básicos es visto como una grilla, pudiéndose allí unir oferentes con sus productos y consumidores esperando utilidad y rendimiento. En base a lo anterior modelos económicos han sido llevados al mundo de las grillas. Las dificultades en la optimización de estos  
dos parámetros en un algoritmo se deben al hecho de que las unidades de medida del coste económico  
y el costo de tiempo son diferentes y porque estos dos objetivos suelen tener conflictos. Por ejemplo,  
recursos proporcionando un mejor rendimiento suelen ser más caros.

El desarrollo de las grillas computacionales ha mostrado una tendencia orientada al servicio y la QoS es otro factor de importancia en las grillas. La presencia de la QoS puede cambiar la selección de los recursos seleccionados.

Los algoritmos de agentamiento de función objetivo centrados en los recursos apuntan a optimizar el funcionamiento del sistema. Estas funciones objetivo son usualmente relacionadas a la utilización de los recursos en el sistema, como rendimiento de trabajo y balanceo de carga.

Como modelos económicos has sido introducidos en el mundo de las grillas, el retronó, que es el beneficio económico que un proveedor de recursos puede obtener por atraer usuarios para que le asignen tareas a sus recursos, es también considerado por el proveedor de recursos.

**PLANEO DE ASIGNACION (Mapping Scheme)**

Hay dos formas básicas de planear las asignaciones, por un lado en la asignación estática se parte que todas las tareas a agendar son conocidas y la información de los recursos también, por otro lado, en la asignación dinámica, su idea principal es que cuando llega la tarea esta se va asignado a los recursos.

**Asignación estática:** cada tarea de un trabajo es asignada a un cierto recurso, por lo anterior una estimación precisa del costo de la colocación se puede hacer antes de la ejecución real, ya q tanto las tareas como la información de los recursos es conocida.

Tiene ventajas como que es de fácil programación desde el punto de vista del agendador y algoritmos heurísticos lo usan como PFAS y pueden dar soluciones cercanas a resultados óptimos incluso para conjuntos de tareas complejos. Pero tiene desventajas en situaciones de fallo de los recursos de la grilla o de que quede aislado del sistema debido a fallas de la red, o son sobrecargados tanto que el tiempo de respuesta llega a ser mayor que el esperado.

Para tratar de solucionar dichos inconvenientes se han introducidos re-agendamiento pero esto trae consigo sobrecarga para re-estimar el nuevo costo y llevar a cabo la migración de la tarea.

**Asignación dinámica:** este tipo de asignación es utilizada cuando el costo de los trabajos es difícil de estimar o cuando los trabajos llegan dinámicamente. Este tipo de agendamiento tiene dos grandes componentes, el sistema de estimación de estado y el de toma de decisiones. El sistema de estimación de estado colecciona el estado de los recursos del sistema de toda la grilla y construye un estimado. Basados en la estimación la tarea es asignada al recurso. En el caso en que el costo de asignamiento individual no esté disponible, un camino intuitivo es tomado para mantener a todo el sistema funcionando adecuadamente es balancear la carga de todos los recursos. La ventaja de el balanceo de carga dinámico es que el sistema no necesita conocer el comportamiento en tiempo de ejecución de la tarea antes de ser ejecutada. Este enfoque es particularmente útil en sistemas que tiene por objetivo primario maximizar la utilización de recursos en vez de minimizar el tiempo de propagación de los trabajos individuales. Si se da el caso de que a un recurso se le asignen muchas tareas, dicho recurso puede iniciar una política de balanceo para decidir que tareas pueden ser migradas a otros recursos.

**ARQUITECTURA DE AGENDAMIENTO**

Hay tres formas de agendar arquitecturas en la grilla: centralizada, jerárquica o distribuida. Los algoritmos de arquitectura centralizada tienen la ventaja ya que pueden utilizar toda la información tanto de los trabajos como de los recursos, la mayoría de los algoritmos están clasificados dentro de esta categoría, ya que son de fácil implementación pero carecen de robustez en la escalabilidad y una pobre tolerancia a fallos.

-Arquitectura jerárquica: es propuesto en donde las particiones del agendador de nivel de grilla central recibe trabajos en múltiples sub-trabajos y asigna dichos sub-trabajos en diferentes agendadores locales distribuidos en diferentes recursos de clústers. Luego los agendadores locales asignaran dichos sub-trabajos a sus recursos disponibles. También carece de una buena escalabilidad y de tolerancia a fallos. Como ventaja tiene que diferentes algoritmos agendadores pueden ser desplegados en el nivel mas bajo acorde a las políticas de administración de recursos.

-Arquitectura distribuida: hay varios agendadores distribuidos sin ningún control central. Se tiene que considerar si los agendadores están trabajando cooperativamente o de forma independiente.

-No cooperativo: en dicho caso los agendadores actúan como entidades autónomas tomando decisiones acordes a sus propios objetivos.

-Cooperativo: los agendadores deben tener políticas locales las cuales confluyan para alcanzar el objetivo común del sistema global.

**ADAPTACION**

En los algoritmos de adaptación el algoritmo en si mismo y sus parámetros pueden variar dinámicamente según el trabajo a realizar y a los estados de los recursos. En la grilla las demandas de adaptación de agendamiento pueden provenir de la heterogeneidad de los **recursos candidatos, de la fluctuación dinámica del rendimiento de los recursos y de la diversidad de las aplicaciones.** Correspondiente a estos tres aspectos tres tipos de algoritmos adaptativos pueden ser definidos.

**Adaptación de recursos:** seleccionar los recursos candidatos es importante ya que de esto depende asegurar su alto desempeño y reducir su costo; por ejemplo las demoras en la transmisión de datos o la velocidad de almacenamiento de los mismos al igual que su capacidad computacional.

**Adaptación de desempeño dinámico:** exhiben estrategias como re-agendamiento o cambio en las políticas de agendamiento, distribución de carga dinámica acorde al desempeño disponible y buscar el adecuado conjunto de recursos para ser seleccionados. Los algoritmos que exhiben estas características usualmente usan el enfoque de dividen y vencerás para resolver un problema, dividiéndolo en varios sub-problemas siendo dicho sub-conjunto a ser creado basado en la disponibilidad de desempeño de recursos.

Adaptación de aplicaciones: alguno agendadores de nivel de aplicación en la grilla están integrados con la aplicación, siendo específicos de la aplicación y de difícil implementación a otras aplicaciones. Superando esta limitación y desacoplando el núcleo del agendador de la aplicación especifica y los componentes de la plataforma especifica, se logra el factor clave, manteniendo al mismo tiempo conciencia de las características, siendo estas descubiertas y recordadas por el compilador especializado y librerías de grilla especializadas. Estas características de las aplicaciones pueden ser obtenidas a través de interfaces bien definidas siendo el agendador de propósito general.

**DEPENDENCIA DE TAREAS**

“Las tareas dentro del mismo trabajo pueden tener relaciones de dependencia o ser independientes de las otras. Usualmente, la existencia de dependencia implica que hay orden de precedencia entre tareas, es decir, algunas tareas no pueden empezar hasta que las otras no hayan terminado”[[13]](#footnote-13).

**AGENDAMIENTO DE CONJUNTO DE TAREAS INDEPENDIENTES**

Cuando no hay relación de dependencia entre tareas, los algoritmos de agendamiento no tienen por qué considerar el orden de asignar las tareas a los recursos, ni la necesidad de considerar el costo de la comunicación de la transferencia de datos de una tarea a otra. Por lo tanto, los algoritmos en esta categoría suelen ser sencillos. Debido a algunas características de la grilla, tales como una alto costo de comunicación, fluctuación dinámica del desempeño, y limitación de funcionalidades en etapas tempranas de la infraestructura de la grilla. Las aplicaciones de grilla fueron usualmente diseñadas con baja granularidad y enviados al meta-agendador por lotes. Por lo tanto, muchos algoritmos de agendamiento de tareas independientes se han aplicado en las grillas.

**AGENDAMIENTO DE CONJUNTO DE TAREAS DEPENDIENTES**

Un enfoque conocido para representar la precedencia entre tareas es el Grafo Acíclico Direccionado DAG (Directed Acyclic graph), en donde sus nodos representan las tareas teniendo un valor numérico que significa el costo computacional, junto a ello, comunicando los nodos están las aristas direccionadas con un valor numérico representado el costo de la comunicación. Con la maduración de la infraestructura de las grillas computacionales y su mejor en cuanto a especificaciones técnicas, flujos de trabajo más complicados son soportados utilizando DAG.

**AGENDAMIENTO DE DATOS**

En el entorno de las grillas el sitio donde se computa la información puede ser desacoplado del sitio donde se almacenan los datos y dichos datos pueden tener varias replicas en sitios diferentes. Por tal razón si solo se tiene en cuenta el costo de computar la información, dicha ganancia va a ser neutralizada por el costo de acceso a los datos. Cinco casos capturan la relación entre los sitios donde se lleva a cabo la computación y los sitios de acceso a los mismos, así:

Datos locales y ejecución local.

Datos locales y ejecución remota.

Datos remotos y ejecución remota.

Datos remotos en la misma ejecución remota.

Datos remotos y en diferente ejecución remota.

Hay dos categorías, cuando la interacción del agendamiento tanto de la capacidad computacional como del agendamiento de los datos es tenido en cuenta, siendo estas categorías.

**Desacoplar el agendamiento de datos del agendamiento computacional**: Hay estrategias de replicación dinámica de datos en la grilla para reducir el consumo de ancho de banda para el acceso de datos remotos como la carga de los puntos cuando se acceden intensivamente. Aunque no es buena estrategia en casos particulares si lo es para ejecuciones prolongadas.

**Produciendo agendamiento de pruebas de computación y de datos**: tienen más consideraciones reales para el mundo de las grillas.

**ESTRATEGIAS DE BUSQUEDA (Mapping search strategies)**

Como en general el problema del agendamiento es difícil y se necesita una confiable asignación de tareas a recursos en un tiempo límite, la mayoría de algoritmos usan heurística de prioridad guiada, que parten de un proceso de pre-agendamiento de las tareas tomando como elementos ya sea estimación de costo, prioridad de jerarquía, ordenamiento u otras.

Por otro lado, hay otro tipo de algoritmos que parten de una asignación aleatoria e iterativamente convergen a una solución de agendamiento buena, asemejando fenómenos naturales, por ejemplo los algoritmos genéticos.

**ESTRATEGIAS PARA MANEJAR LA FLUCTUACION EN EL DESEMPEÑO DE LOS RECURSOS**

Debido a la naturaleza autónoma de los recursos de las grillas y a la competencia entre usuarios por los mismos, estos pueden fluctuar su rendimiento, lo que hace que sea todo un reto para los algoritmos de agendamiento de tareas. Para enfrentar dicho problema se tiene dos estrategias, la predicción de desempeño y re-agendamiento dinámico en tiempo de ejecución.

Predicción de desempeño: “Casi todos los algoritmos estáticos se basan en la predicción de rendimiento para la estimación del costos.” De llagarse a dar una buena estimación se logra un buen agendamiento, pero por la naturaleza incierta del desempeño de los recursos, la falta de exactitud en el desempeño de los recursos y la incertidumbre en su estimación tiene que ser considerada. Teniendo como ejemplo a Takefusa[[14]](#footnote-14) en la arquitectura cliente servidor el tiempo de servicio es calculado por el agendador y si esta estimación no es cumplida el recurso es deshabilitado o el servicio falla debido a la ruptura del plazo.

Schopf utiliza los valores estocásticos basados en la distribución normal para representar comportamientos dinámicos de los recursos de la grilla. En un esfuerzo más reciente se plantea el modelo de cadena de Markov que se utiliza para capturar la dinámica del rendimiento de los recursos;  una cuantificación basada en la entropía del rendimiento de los recursos.

**Re-agendamiento dinámico en tiempo de ejecución:** “El propósito del re-agendamiento es migrar tareas agendadas a recursos alternativos cuando una fluctuación de desempeño sea detectada”[[15]](#footnote-15). En este temática se han introducido conceptos como “contrato de desempeño” que es un acuerdo entre usuarios y proveedores de recursos, cuando un contrato es violado el re-agendamiento es activado.

Hay algoritmos que usan umbral de error para activar el proceso de re-agendamiento. También hay otros algoritmos que el re-agendamiento es solo aplicable a un selecto grupo de tareas para reducir el costo del re-agendamiento. “la idea principal de la póliza selectiva de re-agendamiento es evaluar en tiempo de ejecución la diferencia entre el tiempo de inicio de una tarea contra el tiempo de inicio estimado en el agendamiento estático inicial, solamente las tareas las cuales su diferencia sea mayor al tiempo de retraso permitido, van a ser consideradas para ser re-agendadas”[[16]](#footnote-16).

**PROCESO DE DESICIONES DE MARKOV**

En en capitulo “DEFINICIÓN DE LOS EVENTOS, ACCIONES, ESTADOS Y FUNCIONES DE COSTO DEL PROCESO DE DECISIÓN MARKOVIANO DE TIEMPO CONTINUO” titulo “**Reglas de decisión y política” de este documento se da una breve descripción de los elementos constitutivos de PDM (Proceso de Decisiones de Markov), pero de igual forma se puede consultar el libro de “investigación de operaciones, aplicaciones y algoritmos”, capitulo 19.5, páginas 1036 a la 1045. cuarta edición de Wayne Winston, Editorial Thomson para una descripción amplia de PDM.**

**Modelo arquitectural de la propuesta AG2 (Nivel 2 )**

Vista externa del modelo general de la arquitectura AG2. (Figura)

Vista interna del modelo general de la arquitectura AG2. (Figura)

## 

## Definiciones teóricas (Nivel 2)

Las siguientes son definiciones que se deben dar en la red óptica para tener los elementos necesarios para operar según regla de decisión tomadas por la política estacionaria en proceso de toma de decisiones de Markov.

## Definiciones teóricas del modelo de la red (Nivel 3)

* Se trabajara solo sobre la topología física de fibras y sobre la virtual óptica.
* Se asume que no existen convertidores de longitud de onda (λ) en la red. Esto quiere decir que el λSP ocupa la misma longitud de onda por todas las fibras que pasa.
* Pueden existir multiples λSP´s entres par de nodos.
* Se asume que solo existe una fibra entre los nodos que están conectados.
* Se asume que el camino de menor salto (*minimum hop path*) permanece constante durante el proceso de optimización. Esta presunción es válida porque la adición o eliminación de fibras hace parte de un planeamiento de largo plazo.
* Se asume que la tecnología WDM es empleada para suministrar diferentes longitudes de onda simultáneamente sobre la fibra, y numero de longitudes de onda permanece constante durante el proceso de optimización.
* Se define el ***λSP de nivel 0*** como el LSP **por omisión** que es la representación directa de la fibra en la red física. Por lo tanto cada fibra óptica entre par de nodos contiene un λSP por defecto, no contiene saltos (hops) intermedios y no existe restricción en la longitud de onda a usar. Además los λSP´s de nivel 0 no están sujetos a consideración en el proceso de optimación en lo referente a la creación o eliminación de los mismos, pero si pueden ser objeto de redimensionamiento.

## Definiciones matemáticas del modelo de la red (N3)

Las siguientes definiciones fueron basadas y adaptadas del paper “LSP and λSP Setup in GMPLS Networks”.

Se adicionaran subíndices a las definiciones para expresar que estamos haciendo referencia a un par de nodos, por ejemplo:

1. La red va a estar representada por medio de un grafo (G), compuesto por un conjunto de nodos (***N***) y un conjunto de aristas (L).

***N*** es un conjunto de nodos (conmutadores) en la red y es común para todas las topologías, sean físicas o virtuales.

* 1. Topología de fibras. Donde:
  2. Topología de λSPs:

1. Se define el λSP de nivel cero :
2. Se define un λSP directo como:

*ó eliminar.*

1. Se define la ruta de saltos más corta entre los nodos i y j:
   1. Topología de fibras:
   2. Topología de λSPs:
2. Para cada λSP y Fibra entre nodos i y j, se define la capacidad de ancho de banda:
   1. Topología de fibras:
   2. Topología de λSPs:

.

1. Para cada λSP y Fibra entre nodos i y j, se define la capacidad disponible de ancho de banda:
   1. Topología de fibras:
   2. Topología de λSPs:

.

1. Para cada λSP y Fibra entre nodos i y j, se define la capacidad ya reservada del ancho de banda:
   1. Topología de fibras:
   2. Topología de λSPs:
2. En cada fibra óptica encontramos los siguientes elementos:
3. Solicitudes de ancho de banda que llega a los conmutadores i y j.

## Definiciones teóricas del modelo de la grilla(1):

1. La atomicidad que se tendrá en cuenta será a nivel de **“trabajo”,** por ende, no se contemplaran las **“tareas”** como partes constitutivas del trabajo y tampoco se tendrá en cuenta la dependencia entre tareas, ya que estas últimas simplemente no existen en nuestro modelo.
2. De darse el caso que un trabajo para su procesamiento necesite “**datos en bruto**” (*raw data*), se asumirán que dichos datos en bruto se encontraran en la misma ubicación que la entidad que está generando dichos trabajos.
3. El esquema de asignación de trabajos será de forma “**dinámica**”.
4. La tipo de arquitectura en que los Agendadores (**Middlewares**) se relacionan será la “**arquitectura** **distribuida**” en modo no “**cooperativo**”.
5. Se define como “**Tiempo de propagación**” (Makespam): al tiempo gastado desde el inicio de la primera tarea del trabajo hasta el final de la última tarea del trabajo en el recurso que lo procesa. Por ende, en el marco de nuestro modelo, el tiempo de propagación será el tiempo gastado por el recurso en procesar un trabajo.

## DEFINICIÓN DE LOS EVENTOS, ACCIONES, ESTADOS Y FUNCIONES DE COSTO DEL PROCESO DE DECISIÓN MARKOVIANO DE TIEMPO CONTINUO (2)

Todas las definiciones son provistas para un par de nodos , por lo tanto para la definición formal se suprime el subíndice respectivo.

DEFINICIONES

1. Trabajo:

Podemos definir un trabajo en los siguientes términos:

siendo la ***Cantidad de Datos en Bruto*** que el trabajo necesita para ser procesado y , la cantidad de ***operaciones de punto flotante por segundo*** necesarias para procesar dicho trabajo.

Definir un trabajo en términos que impacten directamente sobre el trafico que va a generar sobre la red y sobre la capacidad de procesamiento que va a consumir en el recurso a donde se asigne es importante porque impacta directamente sobre las partes que constituyen una grilla y por ende se desempeño.

1. Solicitud de ancho de banda.

Se expresa mediante:

especifica la magnitud del ancho de banda solicitado, además de sus puntos de origen y destino.

En la grilla, al asignar una trabajo a un recurso para ser procesado, dicha asignación se traduce en el envió del trabajo con su , lo cual genera sobre la topología óptica un evento directamente relacionado a una .

1. Eventos:

* Se definen los eventos en la grilla:
* Se definen los eventos en la topología óptica:

1. Estados:
   1. Se define el estado de de la topología óptica entre 2 nodos , asi:

Siendo el ancho de banda total disponible en todos los λSP entre los nodos , es la parte de B que esta enrutada en los λSPs directos entre los nodos , es la parte de B que está siendo enrutada en los λSPs sobre y el número de λSPs entre los nodos .

* 1. Se define el estado de la relación entre el cliente y el recurso (c-r), así:

Siendo el número de procesadores libres en un nodo computacional dedicado al procesamiento.

Siendo la capacidad de procesamiento total disponible en un nodo computacional dedicado al procesamiento.

* 1. Se define el estado del conjunto de la topología óptica y la grilla así:

Expandiéndolo, queda así:

1. Se define el **estado extendido** de como , definido así:

Donde son los eventos sobre la grilla y el estado del conjunto de la topología óptica y la grilla.

Lo anterior es pertinente debido a que la única forma para que haga la transición de un estado a otro es mediante la toma de una decisión estacionaria, la cual tomara una acción, y dicha acción, será la que obligue a hacer una transición de estado si es pertinente.

1. Acciones:

Se asume que en el instante de tiempo , el evento ocurre, el cual tiene que ser manejado por la red y la grilla. La red y grilla deciden que acciones llevar a cabo para manejar dicho evento, así:

1. Se definen las acciones en la grilla***:***
2. Se definen las acciones en la topología óptica:

1. **Reglas de decisión y política:**

**“Una regla de decisión provee una selección de una acción para cada estado a un instante de decisión dado . Una política de decisión π especifica las reglas de decisión a ser usadas en el horizonte de tiempo completo donde el problema es considerado, i.e., , donde es el estado. Una política de decisión es llamada estacionaria si la decisión es solamente dependiente del estado de la red y no del instante de tiempo, i.e., para ser más claros para ” (nota al pie)**

1. Función de costo:
2. Definimos el costo sobre la red óptica de un flujo de datos, en tres partes, así:

* El costo de ancho de banda, .
* El costo de señalización, .
* El costo de conmutación,

El costo de ancho de banda como el de conmutación depende del tiempo para el cual el sistema se encuentre en dicho estado. Por lo tanto:

y

,

donde es el tiempo hasta cuando ocurre el siguiente evento. El costo de señalización se da solo en el instante cuando la acción es elegida.

1. Definimos el costo sobre la grilla de procesar un trabajo, así:
2. Se define el costo total de enviar sobre la red óptica flujo de datos, así:
3. Se define el costo total de enviar sobre la red óptica un trabajo junto con sus datos en bruto, procesar dicho trabajo y luego recuperar dichos datos resultantes del procesamiento, así:

donde

Ya que se asume que el “***sistema de estimación de estados***”(3) del agendador, contiene un estado actual y un histórico de los recursos, esta es calculada en un porcentaje donde el 1 significa que el rendimiento estimado y el rendimiento recibido por el recurso fue el mismo. Para <1 se da si el rendimiento recibido es mayor que el rendimiento estimado y >1 si el rendimiento recibido es menor que el rendimiento estimado. Esta se calcula en base al promedio histórico del recurso.**Modelo dinámico de reservación anticipada de recursos de red GMPLS-λSP y de grilla (MDRARG2) (nivel 2)**

Descripción teórica (nivel 3)

La reservación de recursos es una excelente forma de proveer calidad de servicio entre nodos de extremo a extremo e indispensable para aplicaciones de tiempo real sobre una red óptica.

Se debe hacer la clara distinción entre los tipos de reservación; inmediata y anticipa. En la propuesta del analizador multi-dominio se ha abarcado el problema de reservación de recursos de forma inmediata, en esta sección se propondrá un modelo dinámico el cual será responsable de reservar anticipadamente recursos de grilla. Este modelo exclusivamente se enfoca en reservas de tipo STSD[[17]](#footnote-17)(*a specified starting time and a specified duration*) donde el tiempo de inicio y duración es especificado.

“Típicamente las grillas computacionales resultan en transferencias de datos muy grandes entre diferentes sitios, esto puede producir sobrecarga en la redes sino se emplea reservaciones anticipadas”[[18]](#footnote-18).

Beneficios (Nivel 4)

Entre los principales beneficios de la reservación anticipada sobre grillas en redes ópticas se encuentran:

* Los recursos se encuentran disponibles cuando se requieren. Lo cual aumenta la confiabilidad y la robustez de las aplicaciones de grilla.
* La solicitud de recursos no debe esperar tiempo de propagación y de reservación (ej. Tiempo de establecimiento del circuito óptico conmutado **λ**SP ).
* Se podrá estimar con mayor exactitud la duración en la ejecución del trabajo (Makespan), lo cual es ideal en sistemas que poseen límites de tiempo para obtener una respuesta, como los sistemas de tiempo real.
* Se provee un nuevo servicio a la red GMPLS, el cual puede proveer calidad de servicio para los usuarios de la red y “mejorar el planeamiento de las asignaciones de las longitudes de onda de la red tecnología WDM”[[19]](#footnote-19).

Desventajas y desafíos (nivel 4)

Existen grandes retos investigativos, tecnológicos y económicos para afrontar la complejidad de la reservación de recursos en diversos dominios (redes IP, redes ópticas y grillas), tecnologías heterogéneas y múltiples propietarios. Entre los retos más importantes, se pueden encontrar:

* Afrontar las reservaciones anticipadas cuando el tiempo de inicio y/o la duración de la demanda se desconoce. Estos problemas han sido clasificados en STUD, UTSD y UTUD[[20]](#footnote-20). Aunque existen múltiples propuestas de algoritmos para lidiar con esta clase de problemas, las soluciones óptimas resultan típicamente en algoritmos no polinomiales, lo que las hace no viables para su implementación.
* Uno de los principales retos, es soportar la coexistencia de reservaciones anticipadas e inmediatas “para lograr que sean factibles, al diferenciar los recursos de la red de transporte, entre los reservados anticipadamente y los asignados a los usuarios que no pertenecen a la grilla (usuarios estándar ASON/GMPLS”[[21]](#footnote-21)). Para afrontar este reto se requiere un mecanismo de tipo calendario para mantener los recursos coordinados.

Vista general del modelo de reservación de recursos anticipadamente. (figura)

El modelo MDRARG2 está constituido por dos sub-modelos: el analizador de reservaciones y calendario de reservaciones, el primero será el encargado de seleccionar los recursos de grilla que procesaran el trabajo y seleccionara el mejor camino de la red óptica, por lo tanto deberá ser capaz de saber que recursos van estar disponibles en el tiempo de ejecución futuro de la reserva ; el segundo modelo tendrá la responsabilidad de ejecutar la reservación programada para así obtener los recursos necesarios de la solicitud, además tendrá la responsabilidad de evitar posibles conflictos con las reservaciones inmediatas.

**Analizador de reservaciones (Nivel 3)**

Además de los parámetros usados en el analizador multi-dominio como los son:

* Cantidad de *flops* del trabajo,
* Tamaño de la tarea (Row data) y
* Los estados de los caminos de menor salto entre el cliente y el recurso computacional.

Se requieren dos parámetros adicionales que incrementa la complejidad del problema:

* Tiempo en cual se ejecutaran la reservación de recursos.
* Duración de la reserva de los recursos.

Al haber presentado anteriormente la solución del analizador multi-costo, el cual tiene la responsabilidad escoger los recursos de red y de grilla de forma óptica para una petición *T* (vector con los flops y el ancho de banda requerido por el trabajo de la grilla), se presenta la expansión se la solución para adicionar los parámetros adicionales que lograr un mecanismo que reserve recursos de red y de grilla anticipadamente.

**Modelo de reservación *STSD* anticipada para asignación de longitudes de onda[[22]](#footnote-22)(Nivel 4)**

La red pude ser modelado un grafo:

Donde:

*V: es el conjunto vértices que representa los nodos de la red.*

*E: es el conjunto de aristas que represnta los enlaces de fibra.*

*W: es el conjunto de longitudes de onda disponibles en cada fibra.*

Cardinalidad de los conjuntos:

Como los caminos sobre la fibra son conexiones unidireccionales se considera un grafo dirigido donde:

La solicitud de ancho de banda es la capacidad de una longitud de onda sobre la tecnología WDM. Para soportar la reservación anticipada el tiempo es divido en una serie de timeslots de longitud fija, definido asi:

La longitud fija depende de la mínima duración de la reservación anticipada.

El propósito del algoritmo propuesto por Jun Zheng and Hussein y T. Mouftah, es asignar longitudes onda (sin nodos con conversión de longitud de onda) de forma granular sobre un timeslot con una duración de numero entero de timeslot, esto quiere decir que la reservación solo comienza y terminar en un unidad entera de timeslot.

Se define las siguientes funciones y variables:

Para cada enlace

Una serie de vectores , Donde:

Estos vectores son actualizados dinámicamente y se definen así:

Dado el planteamiento anterior, se puede diseñar un algoritmo RWA para el problema de reservación STSD. Se considera un conjunto de rutas de “enrutamiento alternativo fijo[[23]](#footnote-23)” con asignación de longitud de onda aleatoria, donde el conjunto se define:

Y cada ruta:

Donde:

Como la reservación puede únicamente comenzar en un timeslot y la duración es número entero de timeslots, se asume que:

Donde *ks* y *d* son enteros positivos.

**Algoritmo (Nivel 4)**

Se presenta un algoritmo con el objetivo de encontrar una ruta viable que pueda cumplir con los requerimientos específicos.

1. Para cada alternativa ruta se examina la disponibilidad de la longitud de onda para cada enlace a lo largo de la ruta;
2. Para cada enlace , , se examina la disponibilidad de longitud de onda en cada timeslot .
3. Si para cualquier timeslot , se debe ir paso (1) para examinar la siguiente ruta alternativa. En caso contrario examinar si existe una longitud idéntica en el enlace para todo los timeslot, :
4. Si ninguna longitud de onda idéntica está disponible, ir al paso (1), para examinar la siguiente ruta alternativa. De lo contrario examinar si existe una longitud de onda idéntica sobre todos los nodos a lo largo de la ruta durante el tiempo ;
5. Si al menos una longitud de onda idéntica está disponible, una de las longitudes disponibles es seleccionada y reservada sobre cada enlace a lo largo de la ruta durante el tiempo . De lo contrario ir para paso (1) para examinar la siguiente ruta alternativa.
6. Los pasos descritos arriba son repetidos hasta que una ruta viable es encontrada o todas las rutas alternativas son examinas. Si ninguna ruta es encontrada la solicitud será bloqueada.

**Meter diagrmas de flujo pa quede gomelo.**

**Extensión del Modelo de reservación *STSD* anticipada para asignación de longitudes de onda (Nivel 4).**

Se requiere extender el modelo para lograr reservar de forma anticipada los recursos de las grillas, por cual se debe integrar acertadamente el algoritmo anteriormente definido con el algoritmo analizador multi-costo. Con esto se tendría un mecanismo óptimo capaz de escoger y reservar los recursos de red y de grilla de forma anticipada e inmediata.

Inicialmente se presentan las definiciones homologas del algoritmo anterior con las definiciones las de las variables presentadas en el modelo multi-costo.

|  |  |  |
| --- | --- | --- |
| **Descripción** | **Modelos Multi-costo (AG2)** | **Algoritmo de Jun Zheng and Hussein y T. Mouftah** |
| Un grafo (G), compuesto por un conjunto de nodos (***N***) y un conjunto de aristas (L). | G(N,L) |  |
| Nodos | *N* | *N* |
| Aristas | *L* | *E* |
| Cantidad de λs generadas por la tecnología de multiplicación de longitud de onda | *M* | *W* |
| Timeslot de reservación | *No tiene representación\** |  |
| Longitud fija del timeslot | *No tiene representación\** |  |
| Tiempo de inicio especifico | *Tiempo actual.* |  |
| Tiempo de duración específico | *No tiene representación\** |  |
| Tiempo de final específico | *No tiene representación\** |  |
| Disponibilidad de la longitud de onda sobre cada enlace | Pendiente |  |
| conjunto de rutas de enrutamiento alternativo | Conjunto de alternativo |  |
| Ruta de saltos más cortas entre i y j en el grafo G |  |  |
|  |  |  |
| Conjunto de nodos a lo largo de la ruta. |  |  |
| Es conjunto de enlaces directos que conectan a los nodos de la ruta |  |  |
| Enlace unidireccional entre i y j. | *Parte de L en una dirección de i a j* |  |

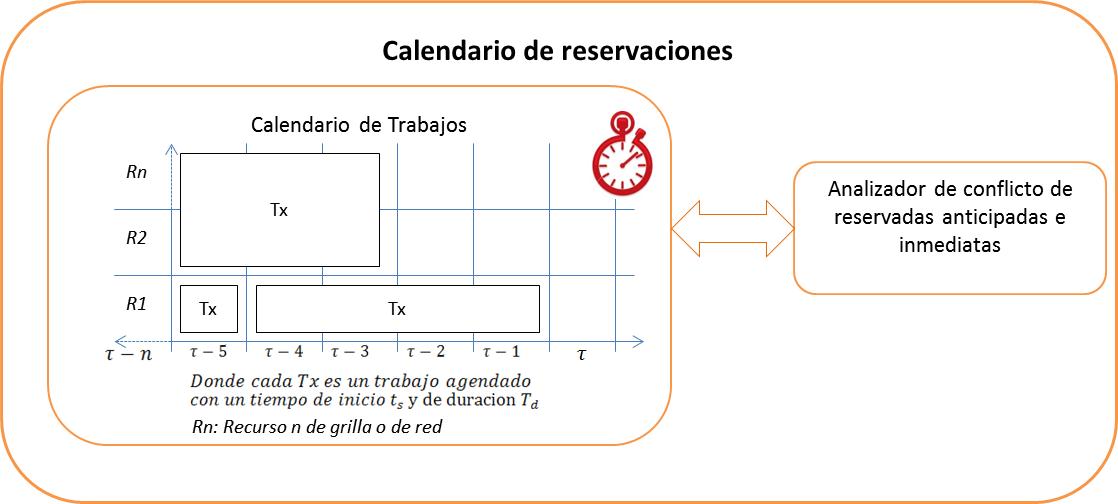
(Tabla)Tabla de homologación de definición de funciones y variables.

\*Se escogen las definiciones planteadas por Jun Zheng and Hussein y T. Mouftah cuando no tengan equivalencia por el modelo multi-costo (AG2).

**Extensión del algoritmo de reservaciones anticipadas. (Nivel 4)**

1. Xxx
2. Xxxx
3. xx

**Calendario de reservaciones** (Nivel 3)



(Figura. Vista general del calendario de reservaciones anticipadas)

**Calendario de trabajos (Nivel 4)**

“Los calendarios deben ser escalables y abiertos para así lograr agendar varios tipo de recursos, incluyendo parámetros de la red (como ancho de banda, identificadores VLAN, timeslot SHD, etc), y además típicos atributos de la grilla (numero de procesadores, cantidad de memoria, etc). El calendario podría ser centralizado en un ambiente multi-dominio, o también podría ser distribuido[[24]](#footnote-24).”

De acuerdo con la extensión del modelo de reservación anticipada y analizador multi-costo planteado anteriormente el calendario debe ser capaz de almacenar la información de las reservas anticipadas como un trabajo para ser ejecutado tiempo futuro cumpliendo con:

* La reserva anticipada se guarda en el calendario como la información del trabajo definida el modelo multi-costo:
* El trabajo es almacenado en el calendario con tiempo de inicio y de duración como números enteros de timeslot.

Donde *ks* y *d* son enteros positivos.

Además de agendar los trabajos el calendario debe guardar información de las reservas de los recursos de la red y de la grilla. Para cada una de las reservas de recursos deberá estar asociada la siguiente información:

Trabajo a ser asignado:

Identificador del recurso:

ID: Identificador único multi-dominio.

Recursos atómicos para ser reservados:

|  |  |  |
| --- | --- | --- |
| **Nombre** | **Definición** | **Tiempos inicio/duración** |
| Unidad de procesamiento masivo |  |  |
|  | Alguna ruta del Conjunto de alternativos. |  |

La reservación de contiene las reservaciones individuales de los recursos de red.

|  |  |  |
| --- | --- | --- |
| **Nombre** | **Definición** | **Tiempos inicio/duración** |
| Enrutadores | Reservación de la conmutación óptica con un costo | ¿?? |
| Enlaces | Reserva de los enlaces sobre una longitud de onda | ¿¡? |

**Analizador de conflicto de reservadas anticipadas e inmediatas (Nivel 4)**

# SIGLAS Y CONVENCIONES (1 Nivel)

|  |  |  |
| --- | --- | --- |
| Sigla | Definición | Definición en ingles |
| ISC | Capacidad de computación de la interface. | Interface Switching Capability |
| LSC | Capacidad de computación de lambda. | Lambda Switch Capable |
| [LSP](http://es.wikipedia.org/wiki/Label_Switched_Path) | Camino conmutado de etiquetas | [Label Switched Path](http://es.wikipedia.org/wiki/Label_Switched_Path) |
| PSC | Capacidad de conmutación de paquetes | Packet Switch Capable |
| SRLG | Grupo de enlaces riesgo compartido | Shared Risk Link Group |
| TDM | Capacidad de conmutación por división de tiempo | Time Division Multiple Capable |
| PCC | Proceso de computación de camino. |  |
| Enlace TE | Enlace de ingeniería de tráfico. | TE Link (Traffic Engineering) |
| MTU | Unidad de transferencia máxima de la interface | Maximum Transfer Unit |
| STSD | Tiempo de inicio y de duración específicos. | A specified starting time and a specified duration |
| FLOPS | operaciones de punto flotante por segundo | floating-point operations per second |
| RWA | Enrutamiento y asignación de longitudes de onda | **routing and wavelength assignment** |

1. Revista disponible internet http://www.acis.org.co/fileadmin/Revista\_107/11.pdf, pag. 11 [↑](#footnote-ref-1)
2. Adrian Farrel, pag 47, cap: LSP Establishment and Maintenance [↑](#footnote-ref-2)
3. RSVP – TE Modeling Specification Message and Data Structure. PDF disponible en: http://snad.ncsl.nist.gov/glass/doc/pdf/gmpls/RSVP\_Message-English.pdf [↑](#footnote-ref-3)
4. http://www.juniper.net/techpubs/en\_US/junos10.0/information-products/topic-collections/nog-mpls-logs/rsvp-objects-structure.html [↑](#footnote-ref-4)
5. PCC (proceso de computación de camino). Acrónimo establecido arbitrariamente para mejorar la lectura del documento. [↑](#footnote-ref-5)
6. Farrel 173 [↑](#footnote-ref-6)
7. Farrel 174 [↑](#footnote-ref-7)
8. Farrel 174 [↑](#footnote-ref-8)
9. Farrel pag 176 [↑](#footnote-ref-9)
10. Farrel 176 [↑](#footnote-ref-10)
11. Farrel 235 [↑](#footnote-ref-11)
12. Farrel 243 [↑](#footnote-ref-12)
13. .WORKFLOW SCHEDULING ALGORITHMS IN THE GRID, Tesis doctoral, Fangpeng Dong, Abril de 2009, pag. 22 [↑](#footnote-ref-13)
14. .A. Takefusa, S. Matsuoka, et al., A Study of Deadline Scheduling for Client-Server, Systems on the Computational Grid, Proc. the HPDC 2001, pages: 406--415, San Francisco, CA USA, August 2001. [↑](#footnote-ref-14)
15. . WORKFLOW SCHEDULING ALGORITHMS IN THE GRID, Tesis doctoral, Fangpeng Dong, Abril de 2009, pag. 54 [↑](#footnote-ref-15)
16. .Ibid. [↑](#footnote-ref-16)
17. Concepto definido en J. Zheng, H.T. Mouftah, Routing and wavelength assignment for advance reservation in wavelength-routed WDM optical networks, in: International Conference on Communications (ICC), 2002. [↑](#footnote-ref-17)
18. Pagina 2989 Contents lists available at ScienceDirect, Computer Networks, Multi-cost job routing and scheduling in Grid networks. [↑](#footnote-ref-18)
19. Pagina 2722(1) J. Zheng, H.T. Mouftah, Routing and wavelength assignment for advance reservation in wavelength-routed WDM optical networks, in: International Conference on Communications (ICC), 2002. [↑](#footnote-ref-19)
20. (STUD). An unspecified starting time and a specified duration (UTSD). An unspecified starting time and an unspecified duration (UTUD). Pagina 2988 Contents lists available at ScienceDirect, Computer Networks, Multi-cost job routing and scheduling in Grid networks. [↑](#footnote-ref-20)
21. Requrimiento de G2MPLS. Pag 5. Phosphorus Grid-enabled GMPLS Control Plane (G2MPLS): Architectures, Services and Interfaces [↑](#footnote-ref-21)
22. Modelo porpuesto en articulo: Routing and Wavelength Assignment for Advance Reservation in Wavelength-Routed WDM Optical Networks [↑](#footnote-ref-22)
23. Enrutamiento alternativo fijo: Donde cada nodo mantiene una tabla de enrutamiento que contiene una ordenada lista de rutas fijas para nada nodo de destino. [↑](#footnote-ref-23)
24. Pagina 5- Phosphorus Grid-enabled GMPLS Control Plane (G2MPLS): Architectures, Services and Interfaces [↑](#footnote-ref-24)