

# Modelo de red interconexión basado en enlace

Máster en Ciencia e Ingeniería Computacional  
Universitat Autònoma de Barcelona  
CAOS - ETSE

Director: Dr. Daniel  
Franco

Alex Ramón Gómez Vera

# Modelo de red Lbased

Introducción

Marco teórico

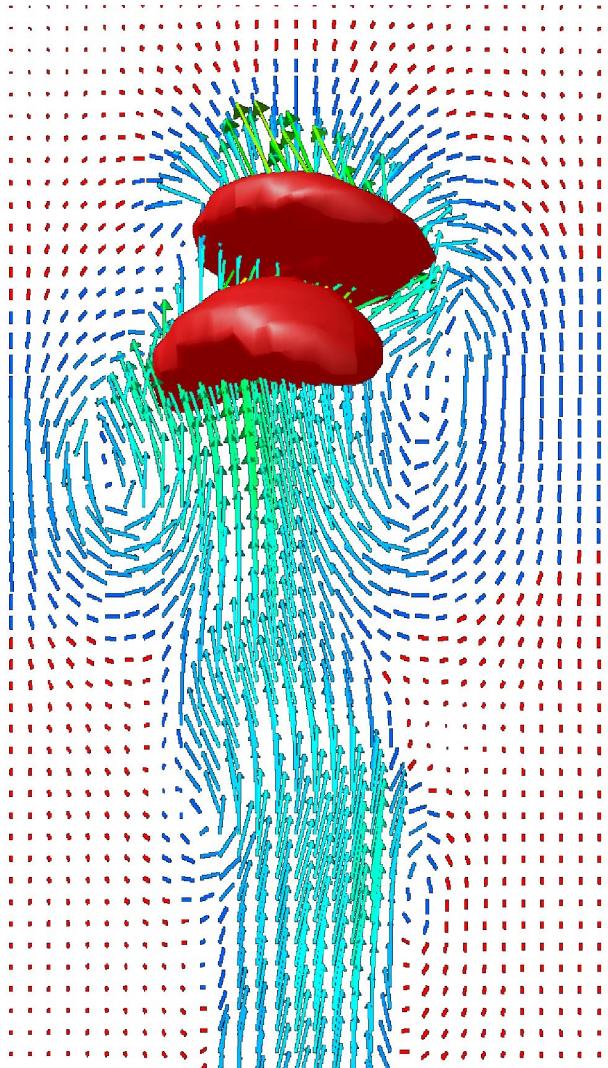
Análisis y Diseño

Resultados

Conclusiones

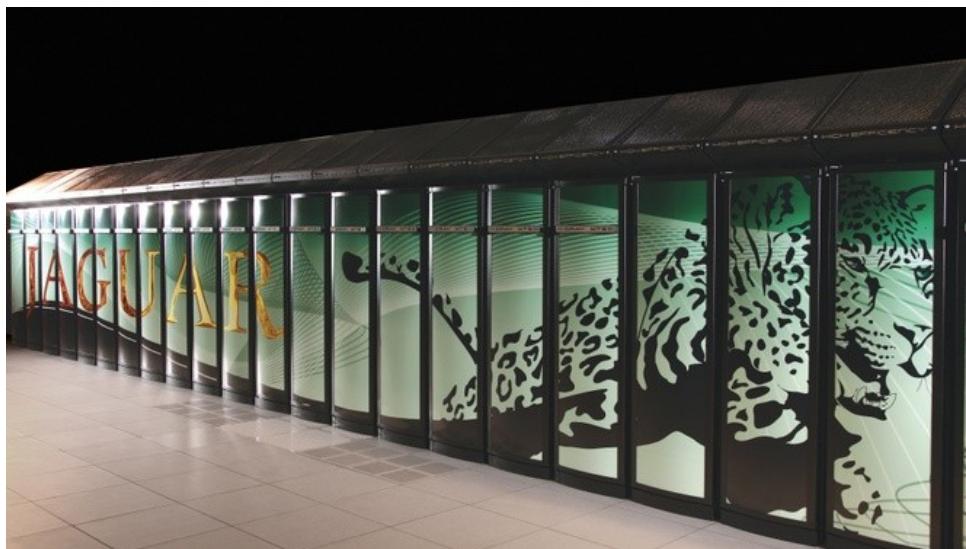
- Introducción
- Motivación
- Objetivo

# Introducción



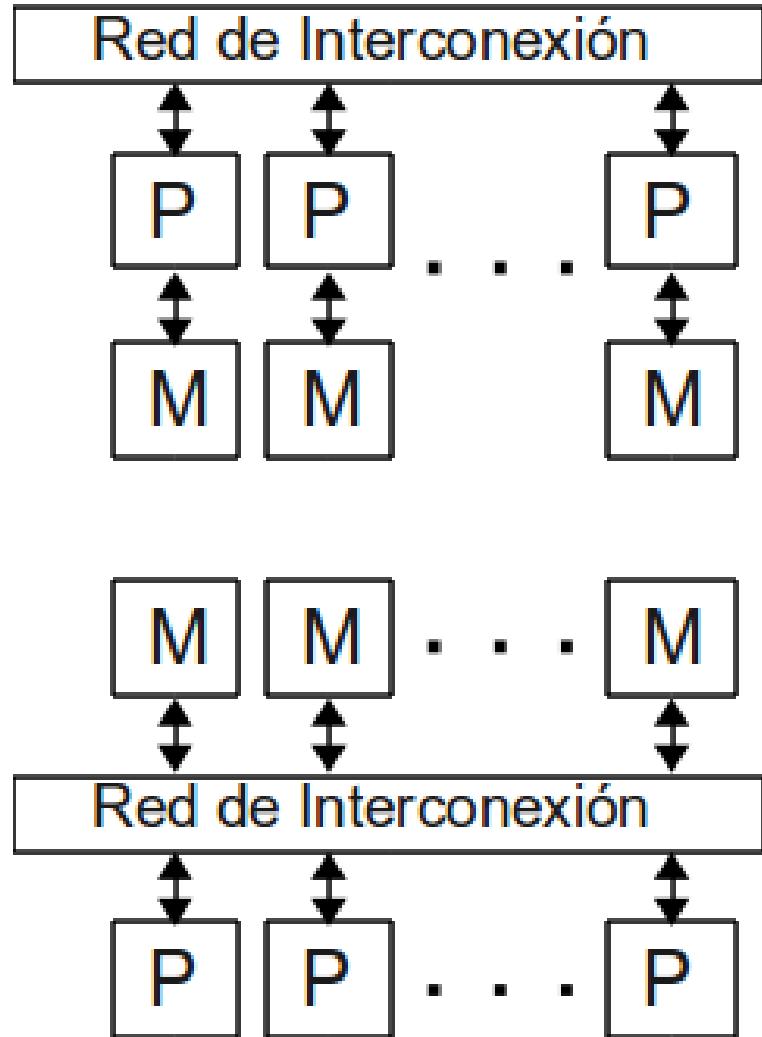
- Hay muchas aplicaciones que requieren mucha capacidad de cómputo, como por ejemplo, las simulaciones de dinámica de fluidos, las simulaciones de ecosistemas y simulaciones de energía nuclear

# Introducción



- En Junio de 2010, se publico que el computador más rápido es el Jaguar, que llega a 1.75 Flops.
- Tiene más de 224000 cores.
- Utiliza una red de interconexión de altas prestaciones.

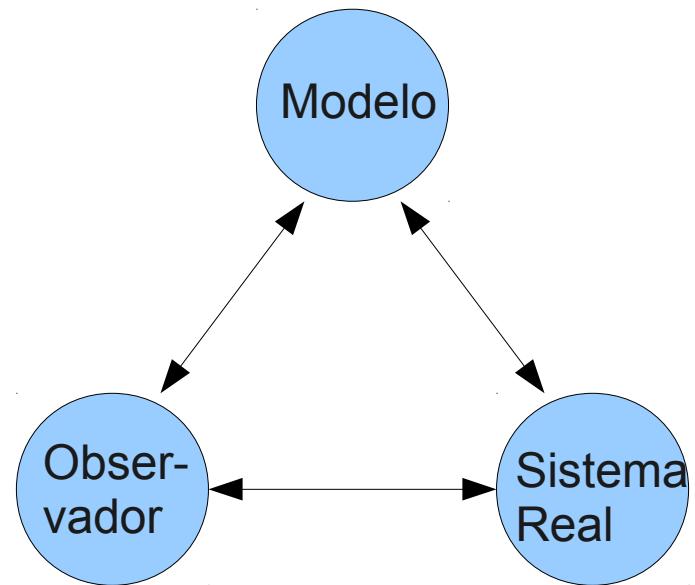
# Introducción



- Tanto en el número 1, como en cualquier computador paralelo masivo, la red de interconexión juega un papel crítico.

# Introducción

- Se requiere hacer un modelo de simulación que permita hacer predicciones, tener un entorno controlado para hacer pruebas y mejorar el conocimiento del sistema.
- Un modelo es una abstracción del sistema.



# Motivación

- Un modelo de simulación ayuda a predecir el rendimiento y a comparar entre diferentes alternativas de diseño.
- Proporciona un método más simple de solución cuando los procedimientos matemáticos son complejos o difíciles.
- Ayuda en el procesos de innovación, ya que permite al experimentador observar y jugar con el sistema.
- Generalmente es más barato mejorar el sistema via simulación que hacerlo en el sistema real.
- No es necesario interrumpir las operaciones de la compañía.

# Objetivo

Las redes de interconexión pueden ser representadas con un modelo que tenga en cuenta algunas de sus variables de estado, se puede optar por diseñar un modelo de simulación que sea muy simple y rápido, o uno muy preciso y lento.

- Diseñar e implementar un modelo que represente el comportamiento dinámico de la red interconexión.
  - Implementar un modelo de simulación que tenga en cuenta la topología y el enrutamiento de la red de interconexión, que sea preciso y rápido.
  - Poder observar los choques producidos por los paquetes que intenten usar un mismo enlace dentro de un intervalo de tiempo dado.
- Entre más simple sea un modelo, más rápida será su ejecución, pero también se aumentará el error.
- Obtener un buen compromiso entre velocidad y precisión.

# Modelo de red Lbased

Introducción
Marco Teórico
Análisis y Diseño
Resultados
Conclusiones

- Sistemas, modelos y simulación
- Redes de interconexión
- Modelos de redes de interconexión
- Modelo de red basado en destino (DBModel)

# Modelo de red Lbased

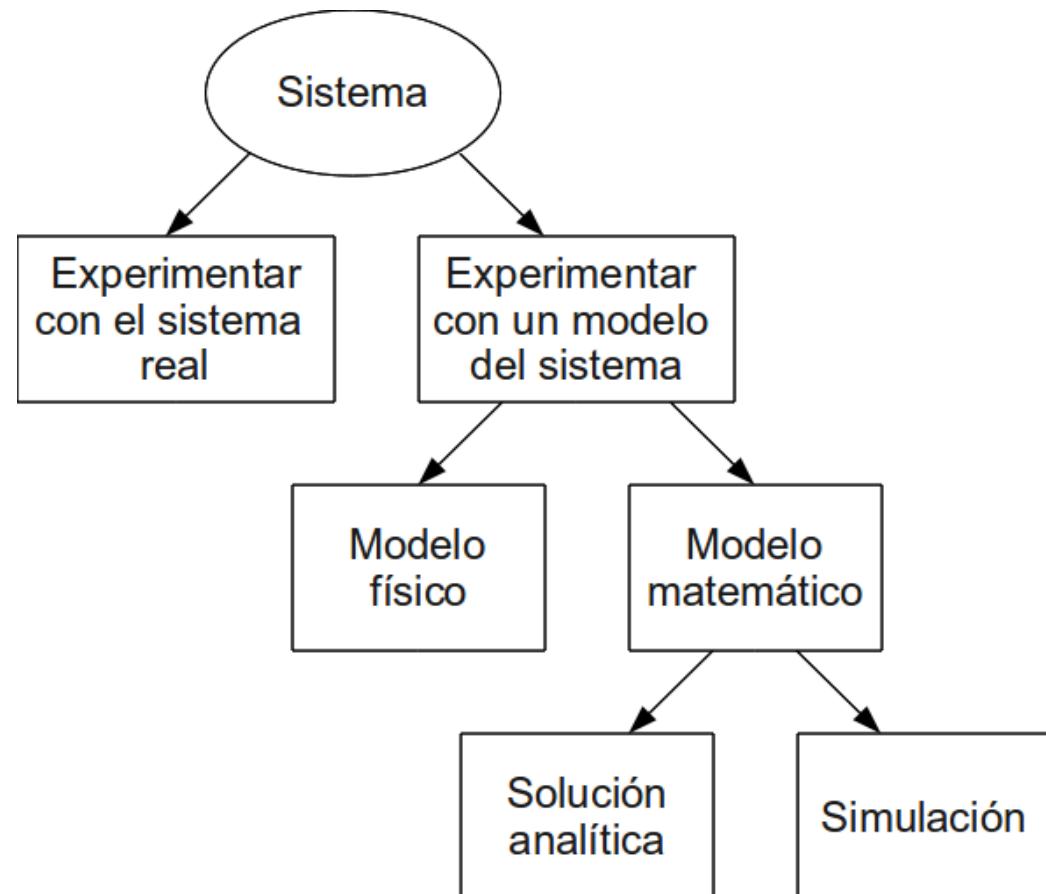
Introducción
Marco Teórico
Análisis y Diseño
Resultados
Conclusiones

- **Sistemas, modelos y simulación**
- Redes de interconexión
- Modelos de redes de interconexión
- Modelo de red basado en destino  
(DBModel)

# Sistemas, Modelos y Simulación

## SISTEMA

- Un *sistema* es definido como una colección de entidades que interactúan entre si para el logro de un determinado fin.

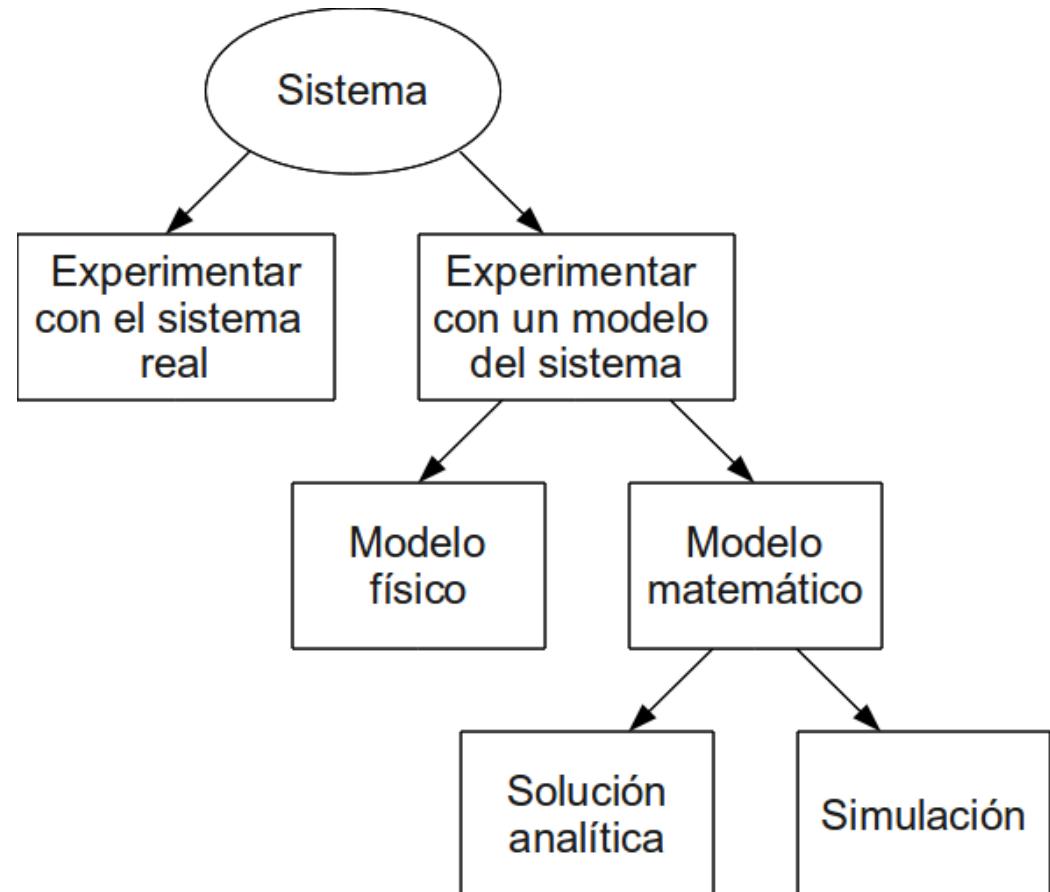


Law & Kelton, "Simulation Modeling and Analysis", 2003

# Sistemas, Modelos y Simulación

## MODELOS

- Los modelos son una abstracción del *sistema* real para ayudar a entender como funciona.

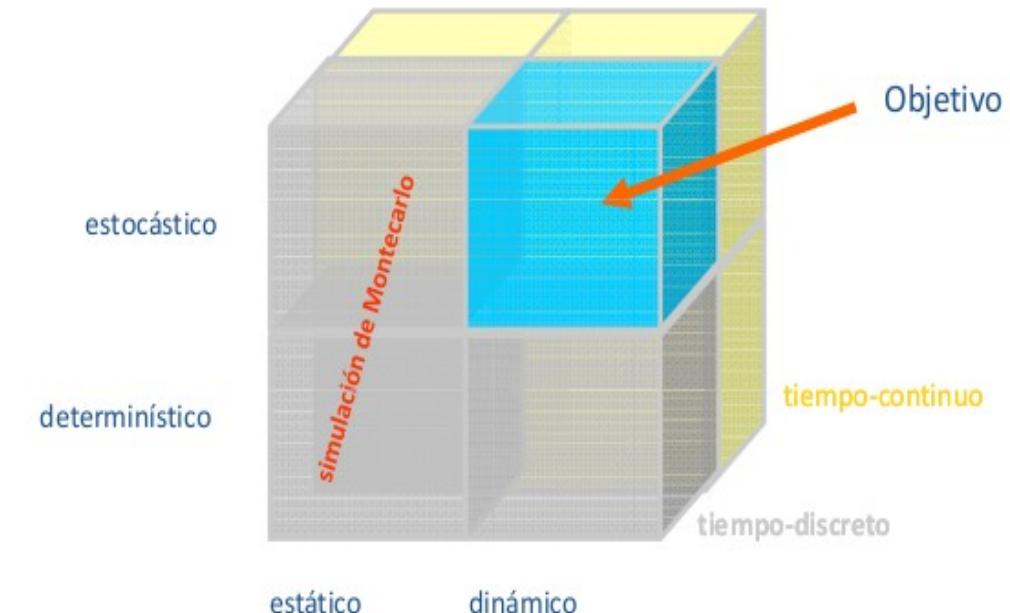


Law & Kelton, "Simulation Modeling and Analysis", 2003

# Sistemas, Modelos y Simulación

## SIMULACIÓN

- Estático vs Dinámico
- Estocástico vs Determinístico
- Continuo vs Discreto
- Los modelos de simulación más ampliamente usados en redes son *discretos, dinámicos y estocásticos*, y normalmente son llamados como modelos de simulación de evento discreto o por su siglas en inglés DES- Discrete-Event Simulation.

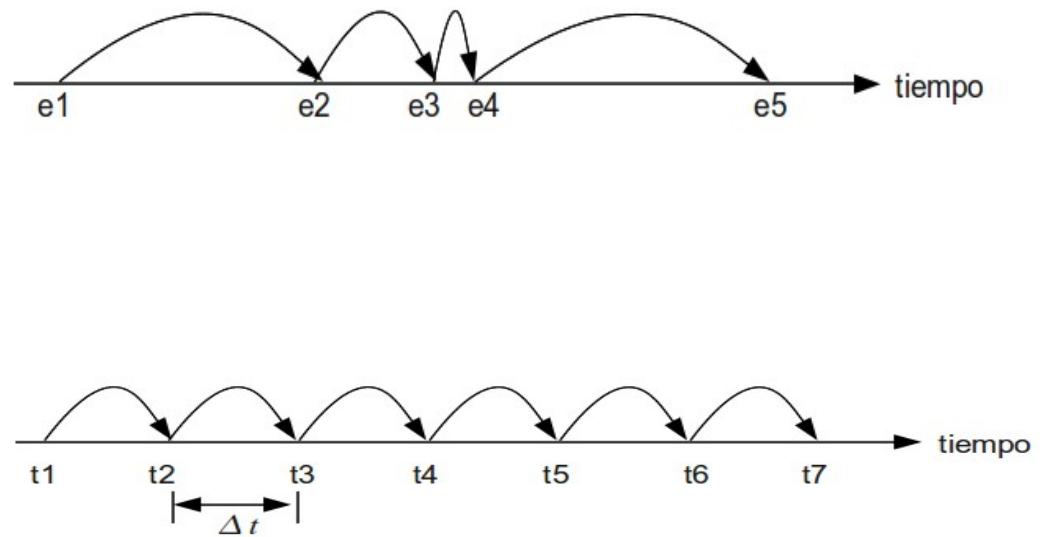


\* Tomado de Clase CSE 2008-2009

# Sistemas, Modelos y Simulación

## SIMULACIÓN DE EVENTO DISCRETO

- Determinar las metas y objetivos.
- Construir un modelo conceptual.
- Convertirlo en una especificación del modelo.
- Convertirlo en un modelo computacional.
- Verificación.
- Validación.



# Modelo de red Lbased

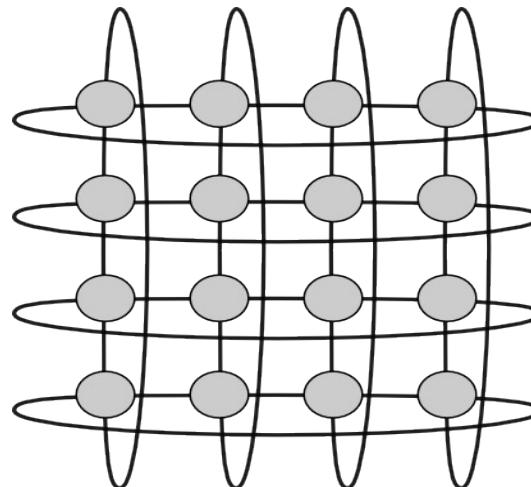
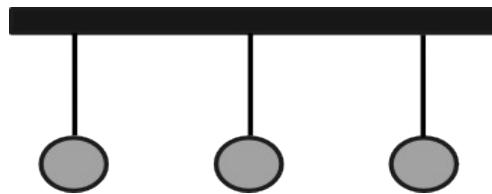
Introducción
Marco Teórico
Análisis y Diseño
Resultados
Conclusiones

- Sistemas, modelos y simulación
- Redes de interconexión
- Modelos de redes de interconexión
- Modelo de red basado en destino (DBModel)

# Redes de interconexión

## TOPOLOGÍAS DE RED

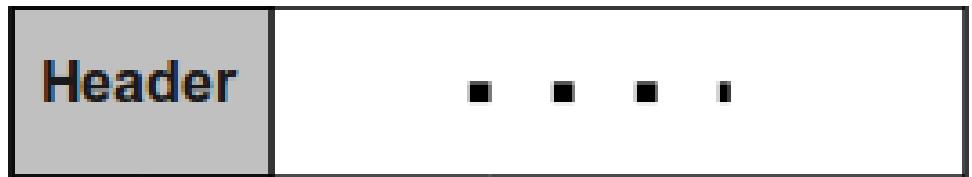
- Redes de medio compartido
- Redes directas
- Redes indirectas



# Redes de interconexión

## TÉCNICAS DE COMMUTACIÓN

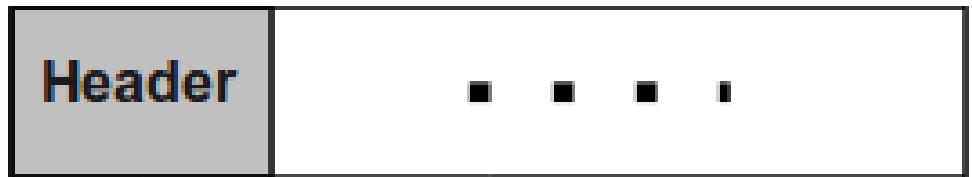
- Comutación VCT
- Comutación Store and Forward
- Comutación Wormhole.



# Redes de interconexión

## TÉCNICAS DE COMMUTACIÓN

- **Commutación VCT**
- Commutación Store and Forward
- Commutación Wormhole.



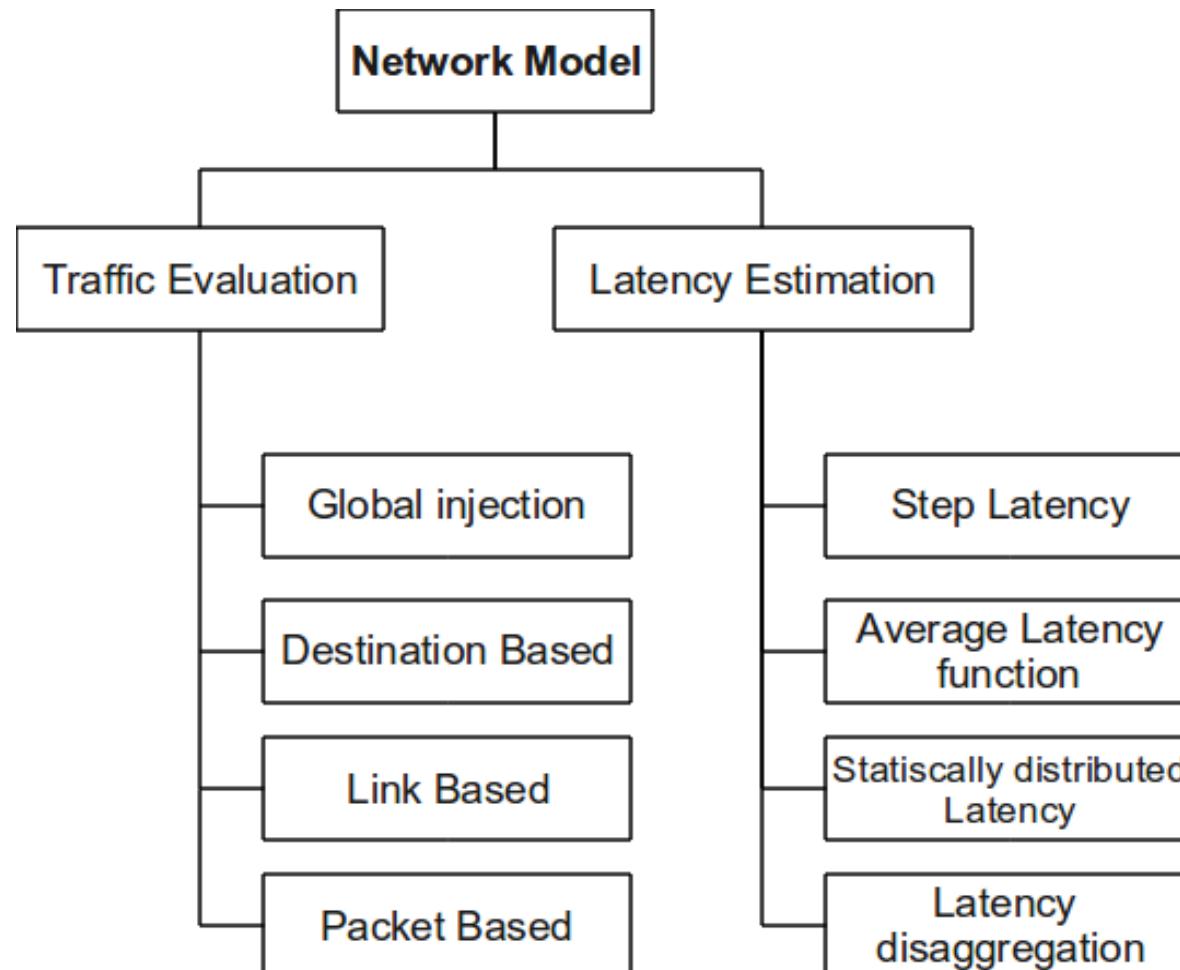
# Modelo de red Lbased

Introducción
Marco Teórico
Análisis y Diseño
Resultados
Conclusiones

- Sistemas, modelos y simulación
- Redes de interconexión
- Modelos de redes de interconexión
- Modelo de red basado en destino (DBModel)

# Modelos de redes de interconexión

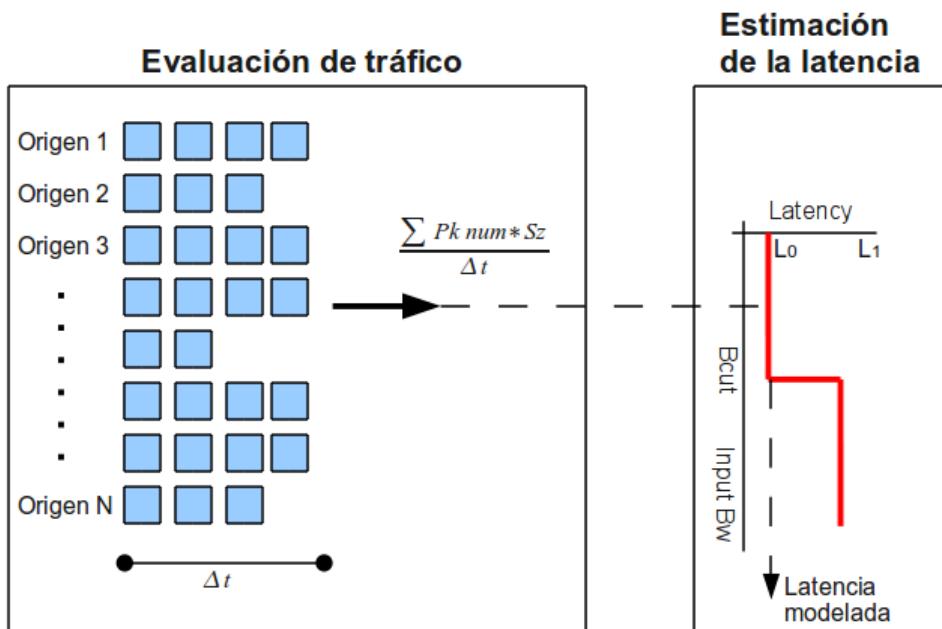
- Espacio de diseño de los modelos de redes de interconexión



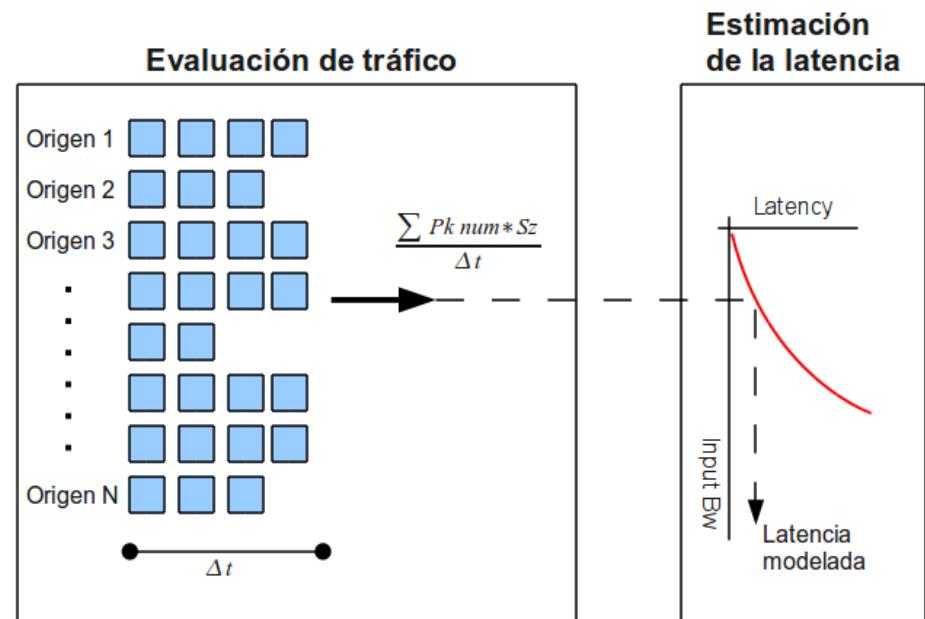
# Modelos de redes de interconexión

## GLOBAL

- Step

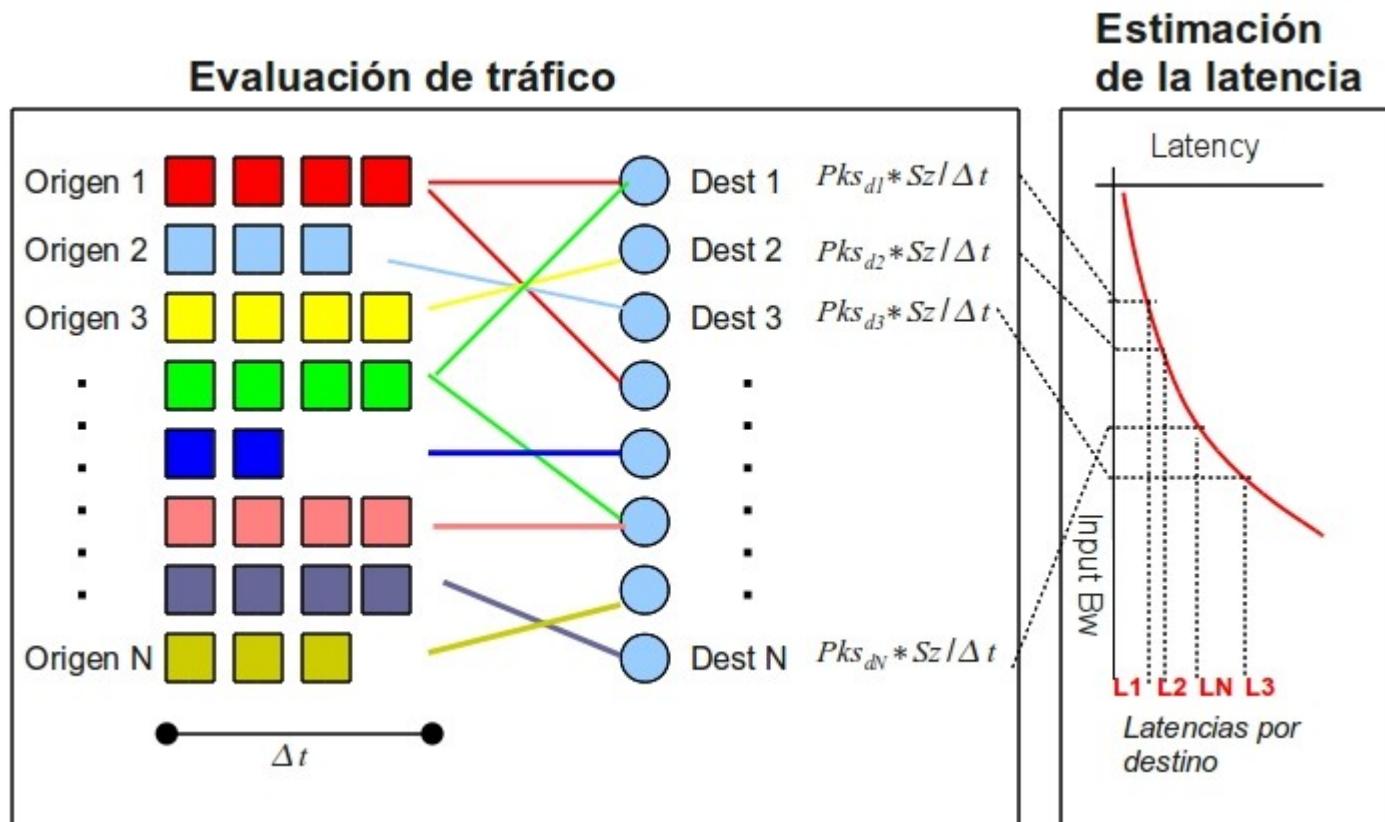


- Average



# Modelos de redes de interconexión

## BASADO EN DESTINO



# Modelo de red Lbased

Introducción
Marco Teórico
Análisis y Diseño
Resultados
Conclusiones

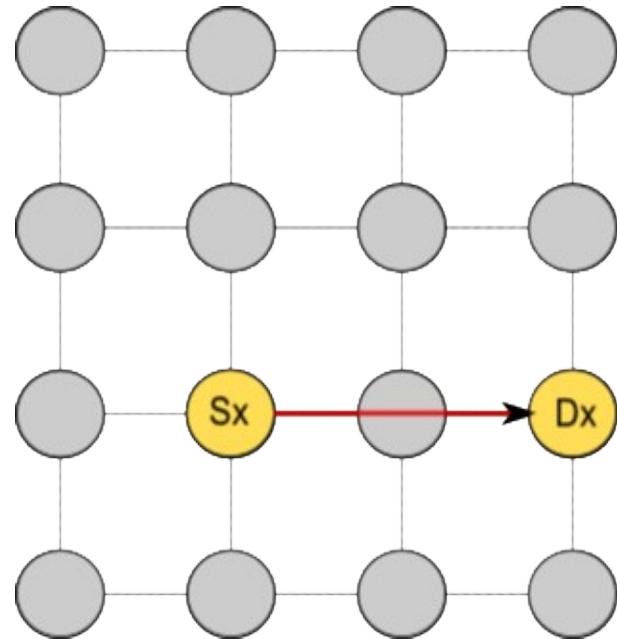
- Análisis del modelo de red basado en destino (DBModel)
- Diseño del modelo basado en enlace (LBModel)

# Modelo de red Lbased

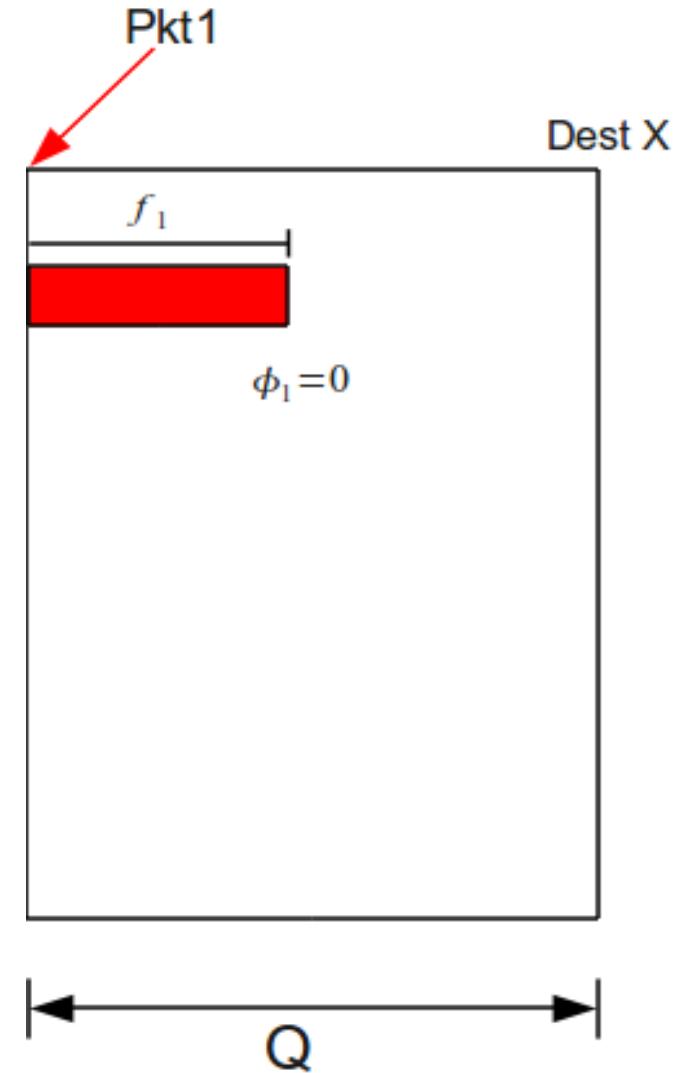
Introducción
Marco Teórico
Análisis y Diseño
Resultados
Conclusiones

- Análisis del modelo de red basado en destino (DBModel)
- Diseño del modelo basado en enlace (LBModel)

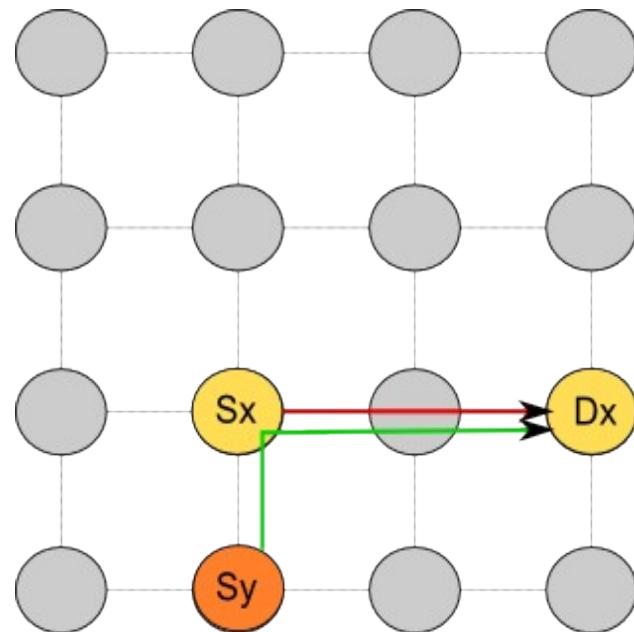
# Análisis del DBModel



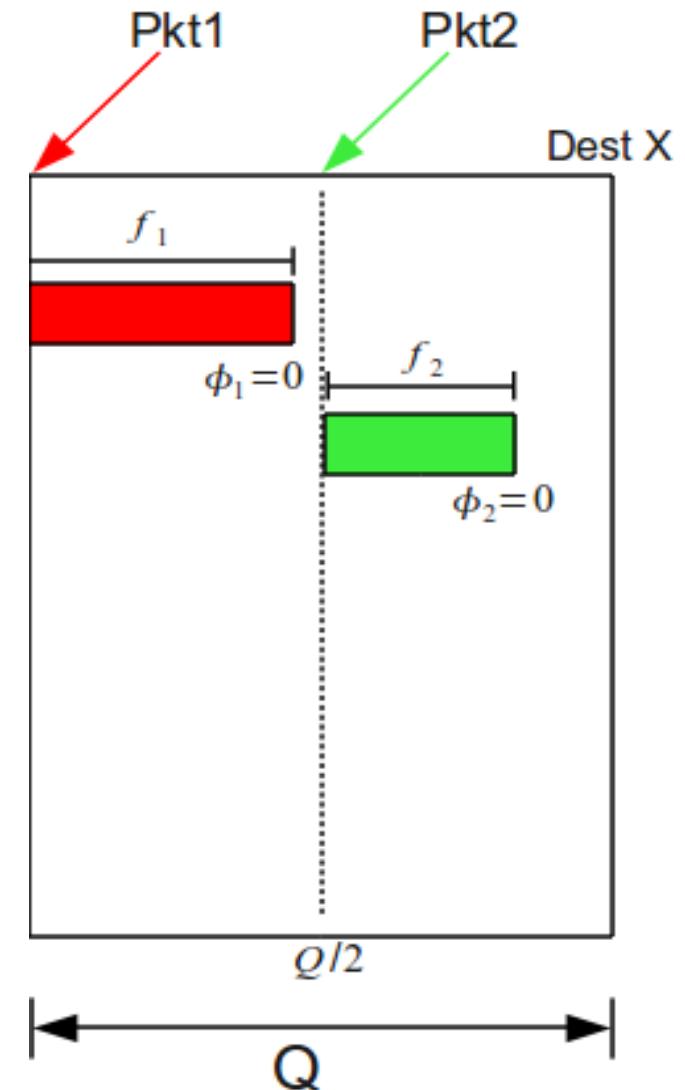
Cola Dest X



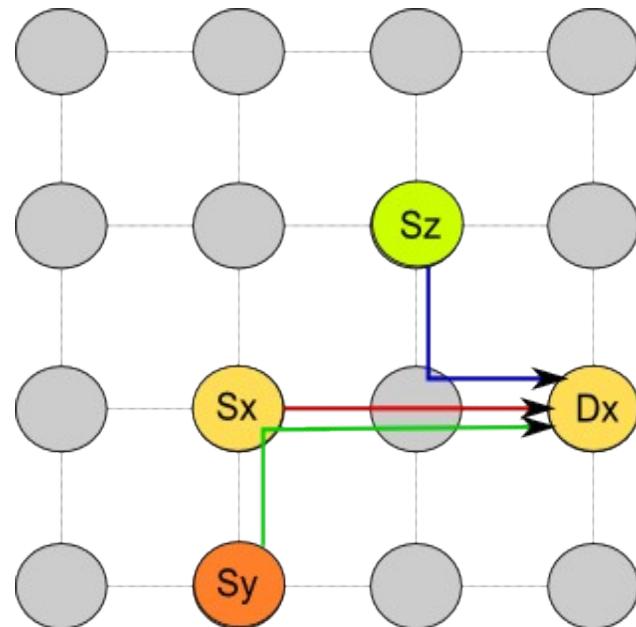
# Análisis del DBModel



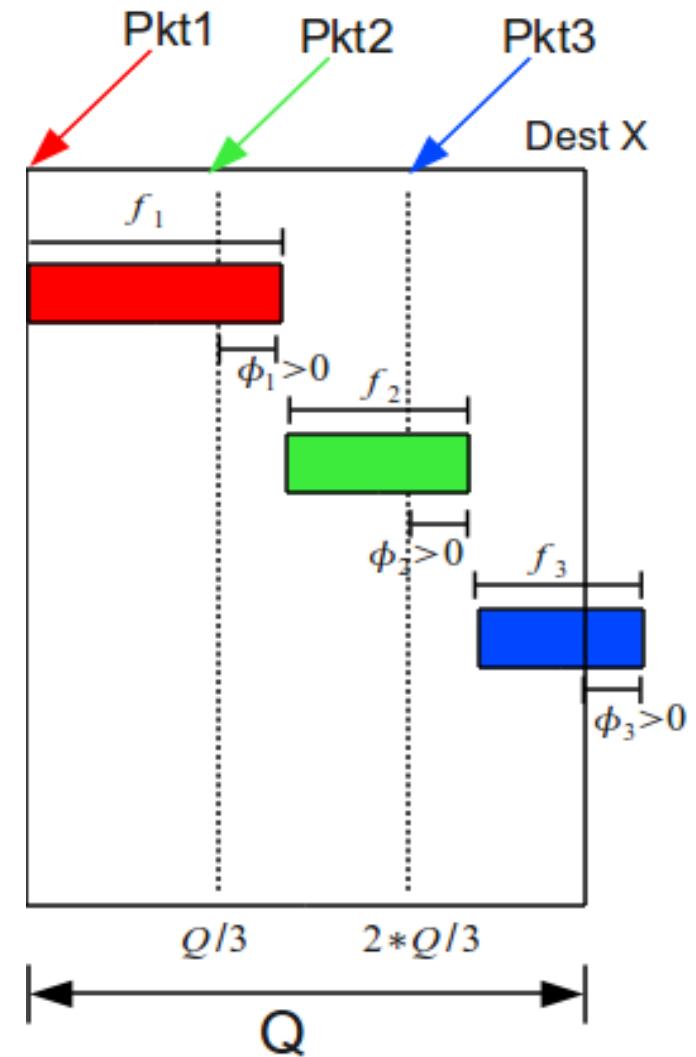
Cola Dest X



# Análisis del DBModel



Cola Dest X



# Análisis del DBModel

## PROCEDIMIENTO DBMODEL

- Identificar el nodo destino
- Calcular la distribución de arribo incluyendo el paquete nuevo.
- Calcular la contención  $\Phi$  para la cola de destino correspondiente.

$$\begin{aligned}\phi[i-1] &= \phi[i-2] + f[i-1] - \frac{Q}{Npks} & \forall i > 0, y \phi > 0 \\ \phi[i-1] &= 0 & \forall \phi < 0\end{aligned}$$

# Análisis del DBModel

## PROCEDIMIENTO DBMODEL

- Calcular la latencia para el i-ésimo paquete.

$$Lat[i] = f_{Netcomp}(N)[i] + \phi_{QueueDynamics}[i-1]$$

# Modelo de red Lbased

Introducción
Marco Teórico
Análisis y Diseño
Resultados
Conclusiones

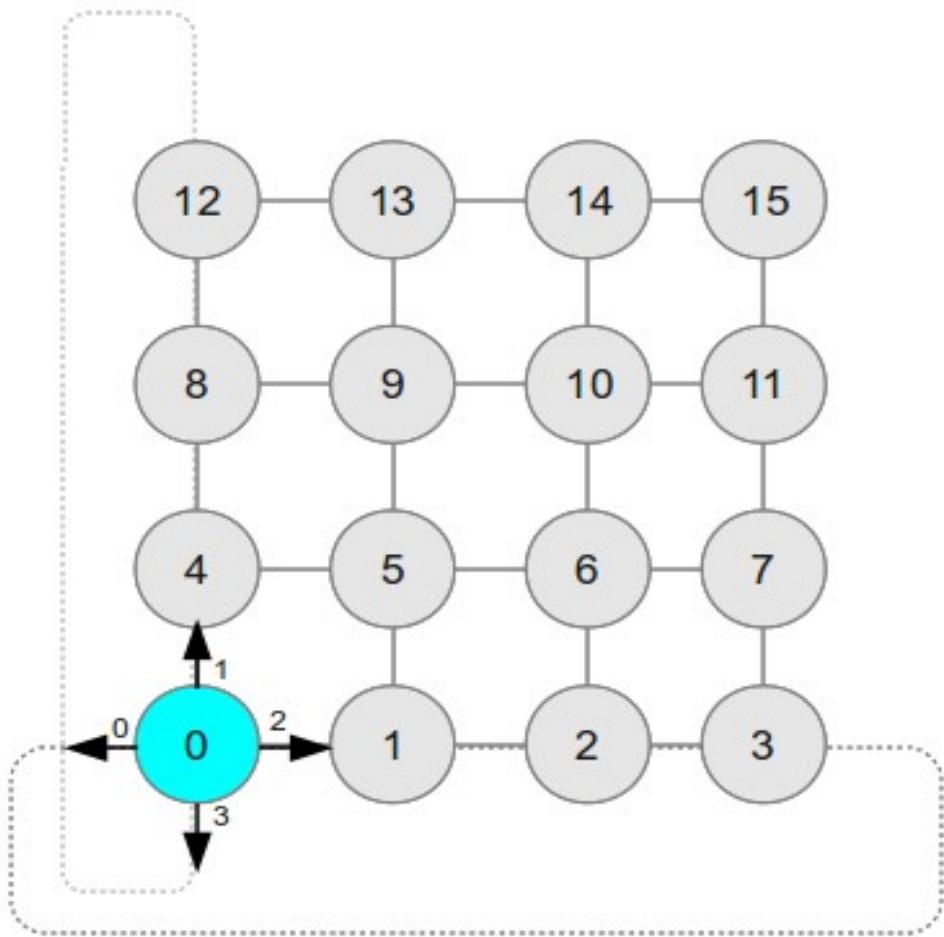
- Análisis del modelo de red basado en destino (DBModel)
- Diseño del modelo basado en enlace (LBModel)

# Diseño del LBModel

- Notación de los enlaces

El número de enlace es dado por numero de nodo agregando al final los dos bits correspondientes.

Ej: 0001 11, el número de link seria 8.

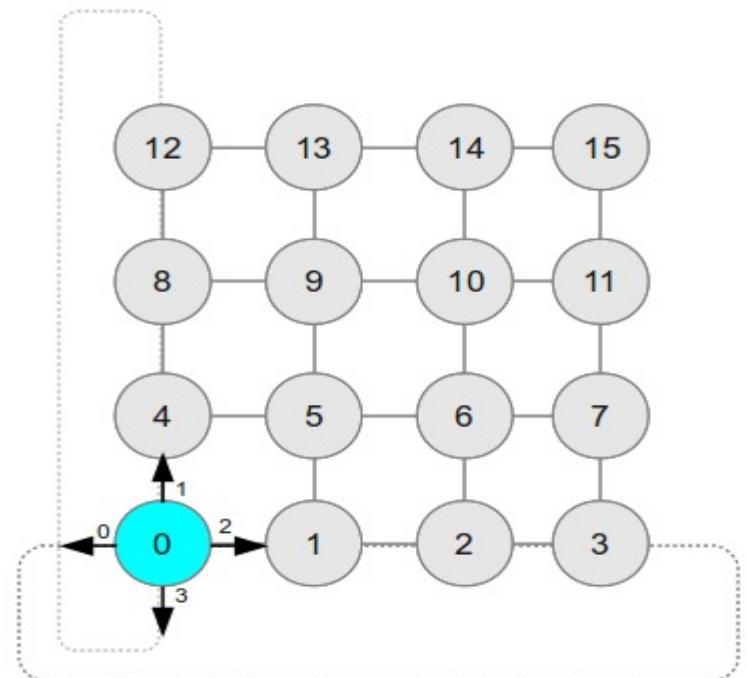


# Diseño del LBModel

## TABLA DE TOPOLOGIA

- Se tiene el nodo Origen y el enlace, y se busca el nodo Destino.

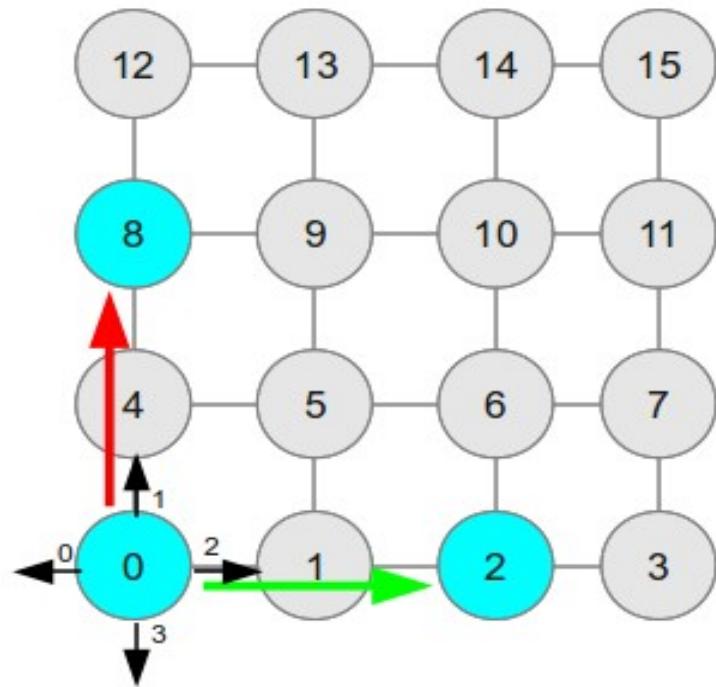
Enlace	Origen	0	1	...	15
0		3	0	...	14
1		4	5	...	3
2		1	2	...	12
3		12	13	...	11



# Diseño del LBModel

## ALGORITMO DE ENRUTAMIENTO XY

- El paquete irá primero por el eje X y luego por el eje Y.

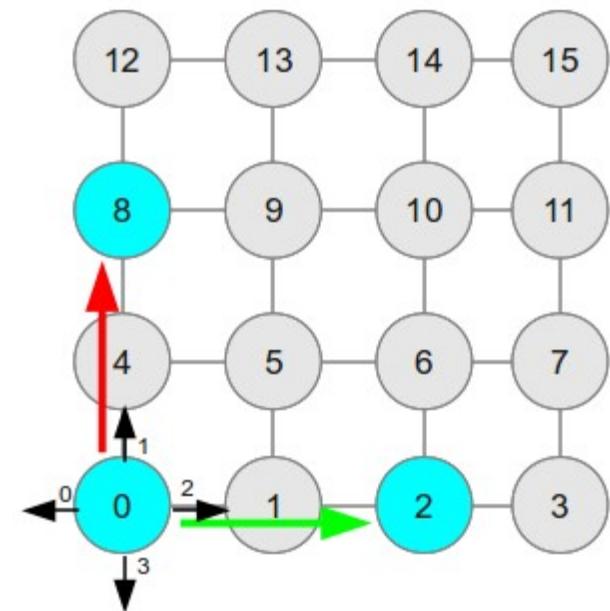


# Diseño del LBModel

## TABLA DE ENRUTAMIENTO

- Se tiene el nodo Origen y el nodo Destino, y se busca el enlace.

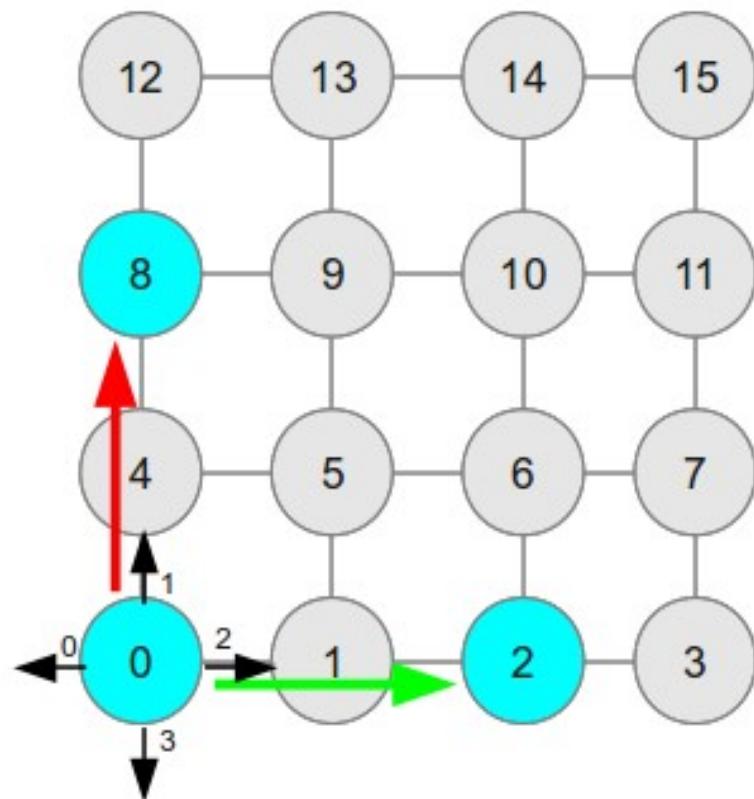
Origen \ Destino	0	1	2	...	15
0	—	2	2	...	0
1	0	—	2	...	2
2	0	0	—	...	2
:	:	:	:	..	:
15	2	0	0	...	—



# Diseño del LBModel

## PROCEDIMIENTO PARA LISTADO DE ENLACES

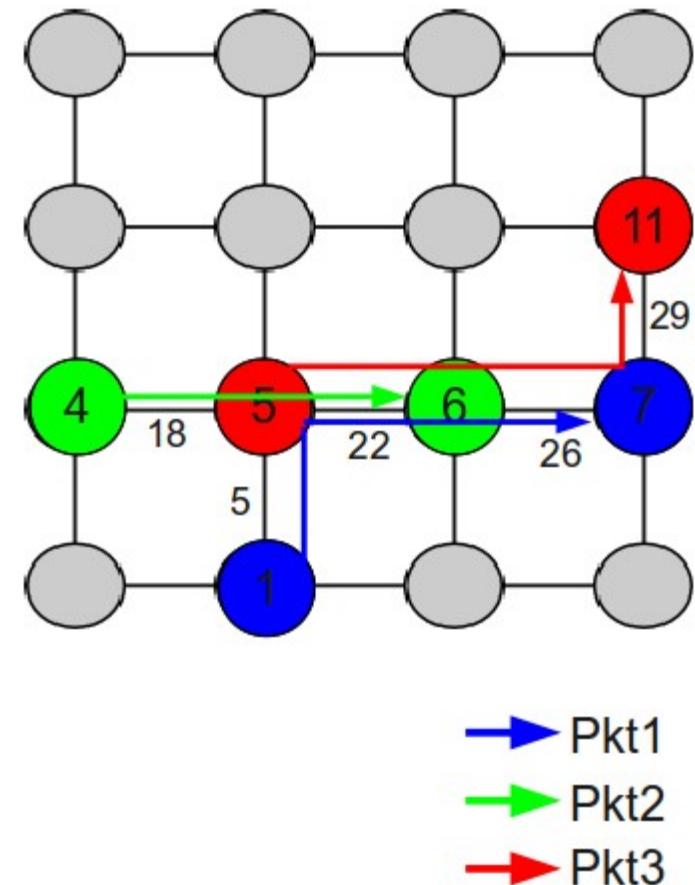
- Entradas: src, dst.
- Consultar routing table[src][dst]-->link
- Agregar link al link-list
- Consultar topology table[link] [dst]-->srctmp
- Iterar hasta que src=srctmp
- Salida: Link-list  
(list STL)



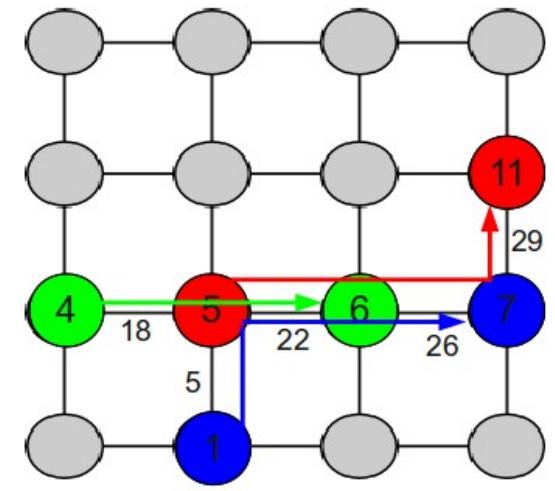
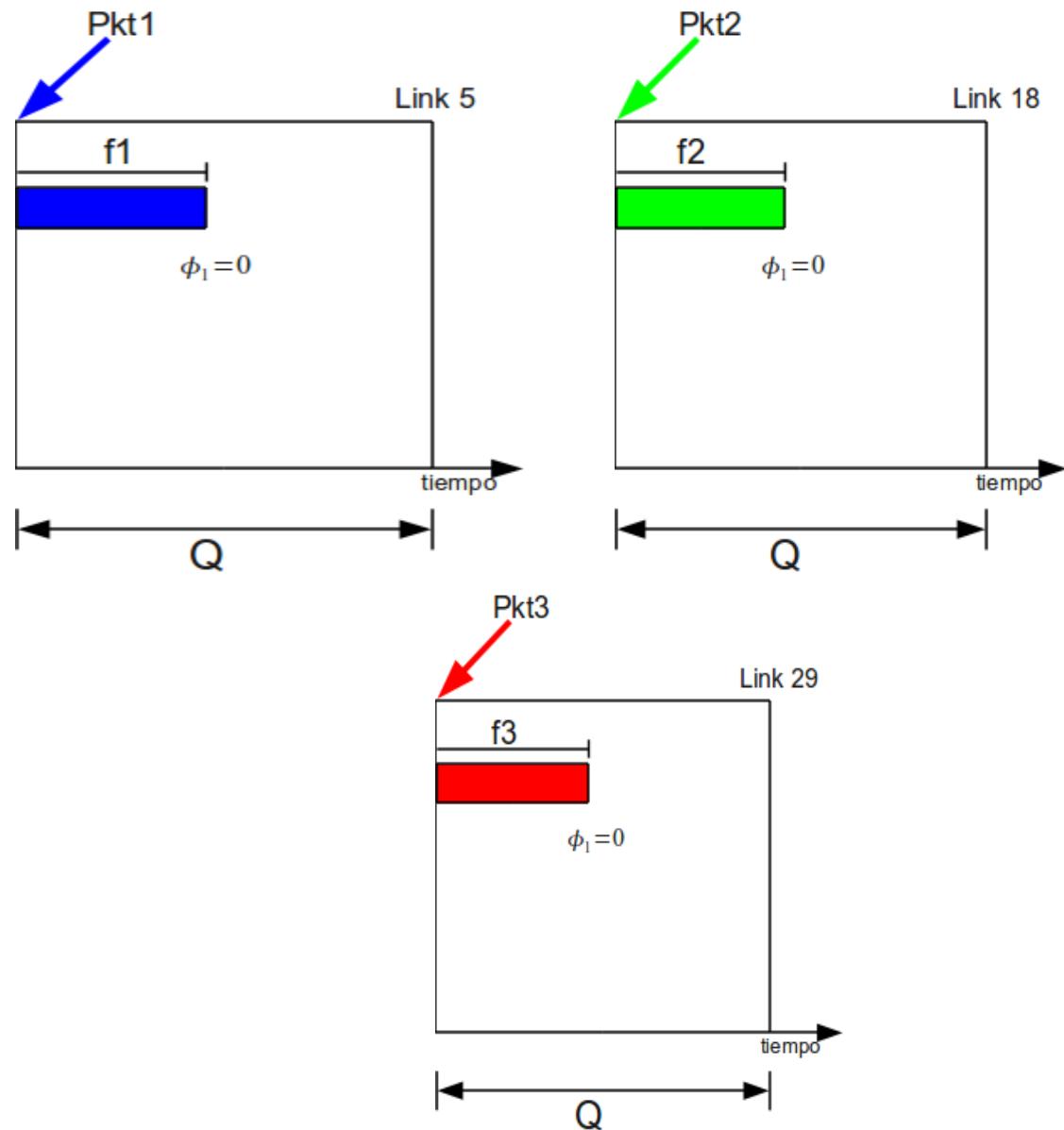
# Diseño del LBModel

## APROXIMACIÓN AL MODELO

- Pkt1:=links[5,22,26]
- Pkt2:=links[18,22]
- Pkt3:=links[22,26,29]
- Cada link maneja un cola independiente.

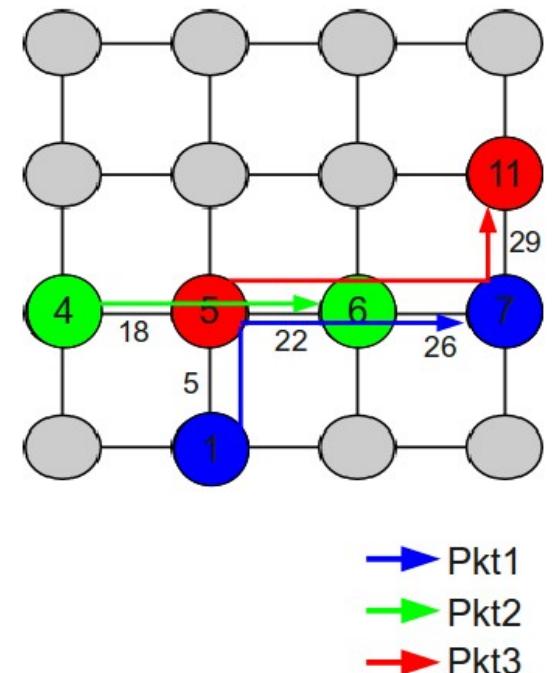
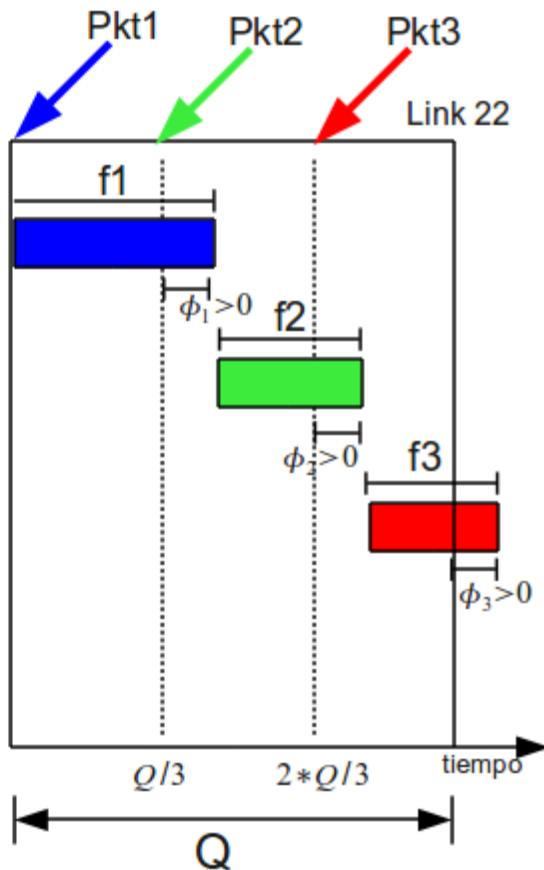
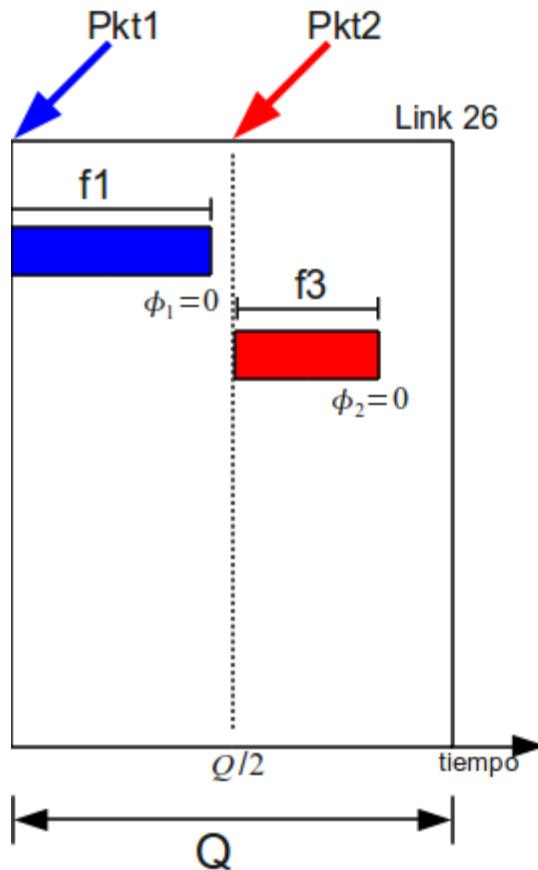


# Diseño del LBModel



- Pkt1
- Pkt2
- Pkt3

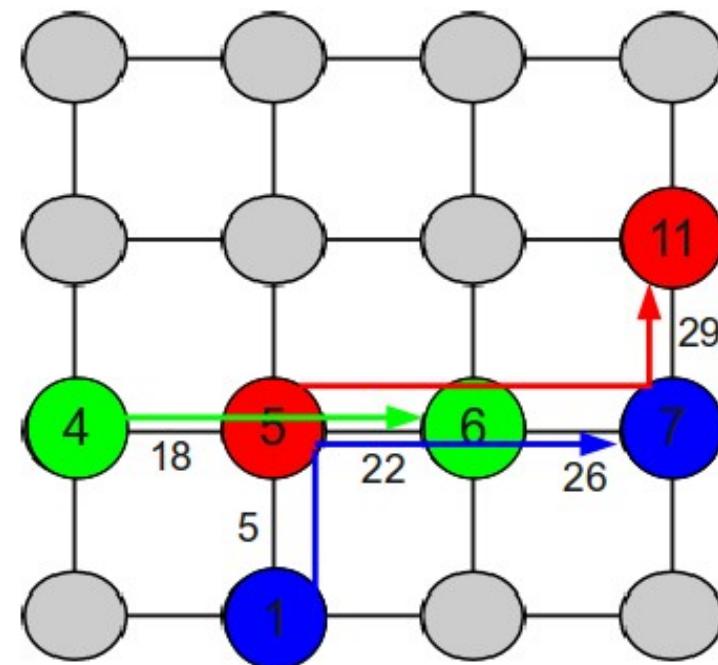
# Diseño del LBModel



# Diseño del LBModel

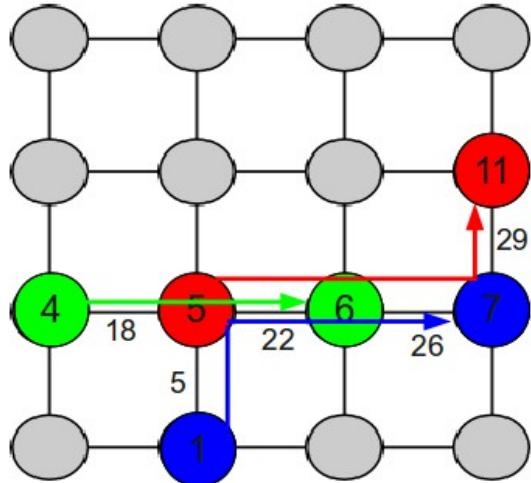
- Pkt1, tendrá distintas contenciones en cada uno de sus links.
- Igual para Pkt2 y Pkt3.

Cual contención se debe considerar?



- Pkt1
- Pkt2
- Pkt3

# Diseño del LBModel



- > Pkt1
- > Pkt2
- > Pkt3

$$\text{Latencia}_{Pk} = F(\text{Lat}_{link_1}, \dots, \text{Lat}_{link_n})$$

$$\text{Latencia}_{Pk} = f(N) + F(\phi_{link_1}, \dots, \phi_{link_n})$$

$$\text{Latencia}_{Pk} = f(N) + \max(\phi_{link_1}, \dots, \phi_{link_n})$$

# Diseño del LBModel

## PROCEDIMIENTO LBMODEL

- Identificar los links por los que pasa el paquete.
- Calcular la distribución de arribo para cada link, incluyendo el paquete nuevo.
- Calcular la contencion  $\Phi$  para cada cola de link correspondiente.

$$\begin{aligned}\phi[i-1] &= \phi[i-2] + f[i-1] - \frac{Q}{Npks} & \forall i > 0, y \phi > 0 \\ \phi[i-1] &= 0 & \forall \phi < 0\end{aligned}$$

.

# Diseño del LBModel

- Calcular la contención máxima entre los links por los que pasa el paquete
- Calcular la latencia para el  $i$ -ésimo paquete

$$Lat[i] = f_{Netcomp}(N)[i] + \phi_{QueueDynamics}[i-1]$$

# Modelo de red Lbased

Introducción

Marco Teórico

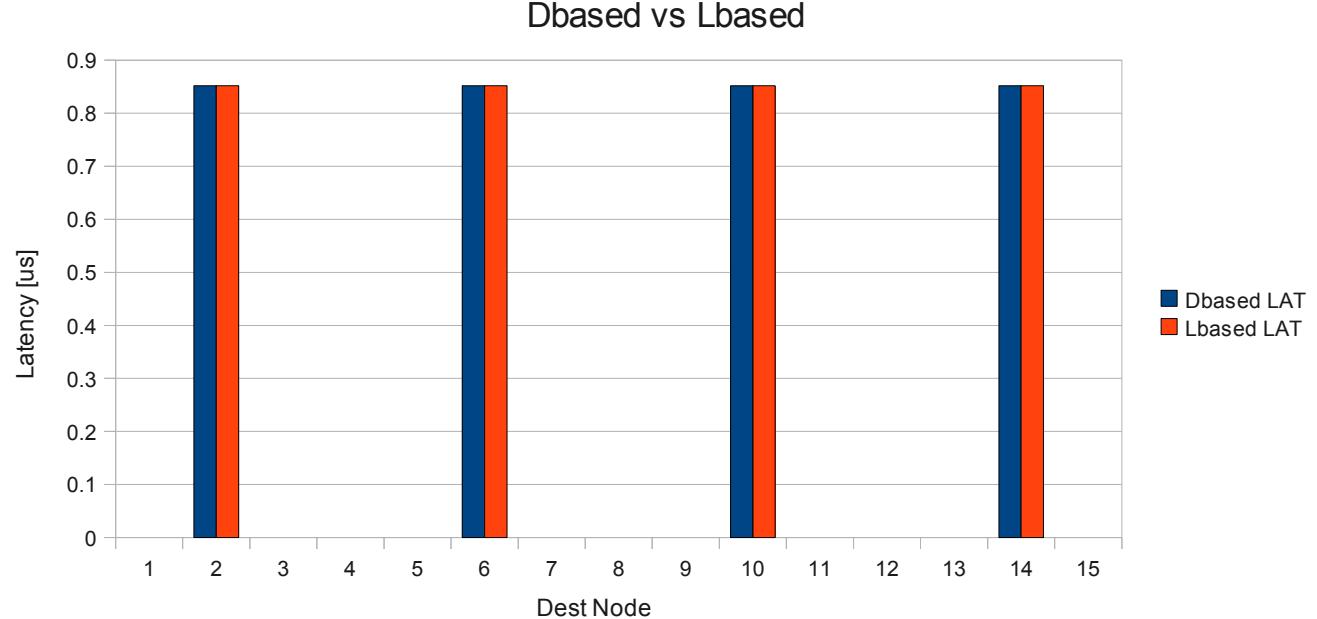
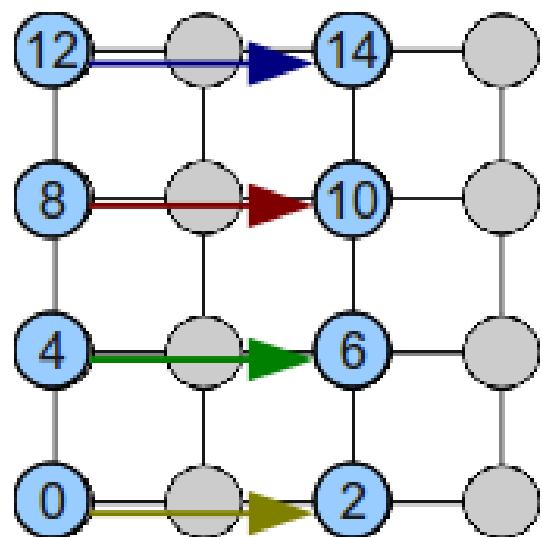
Metodología

Resultados

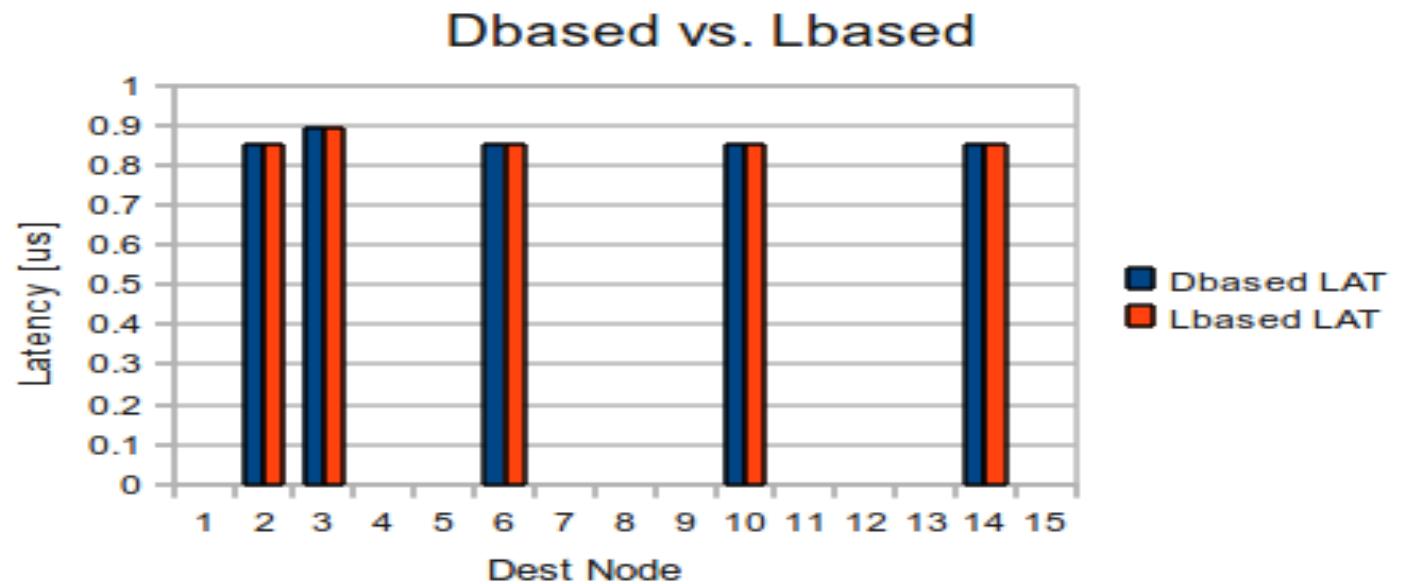
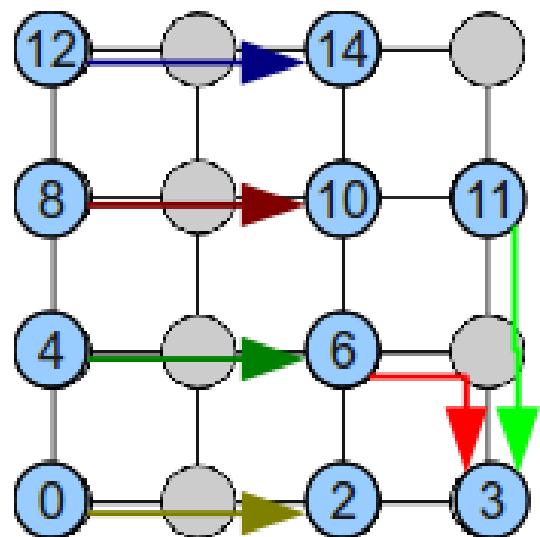
Conclusiones

- Comparación con el modelo Dbased

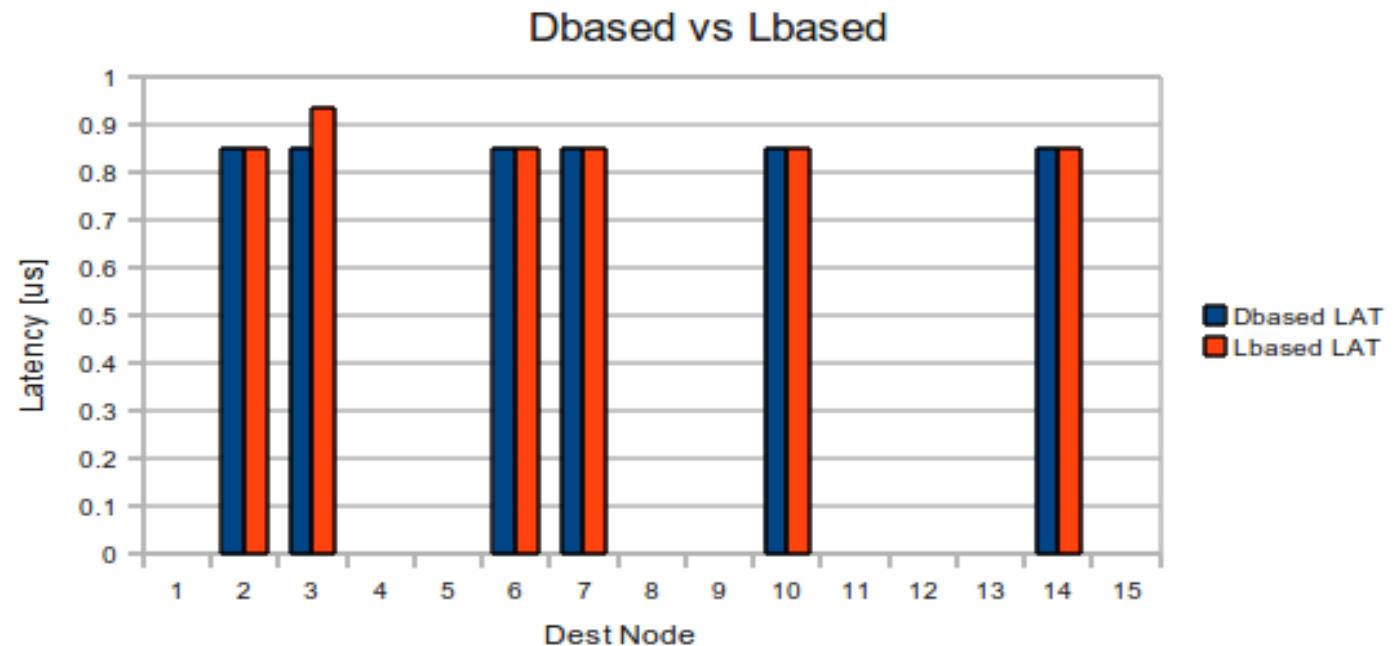
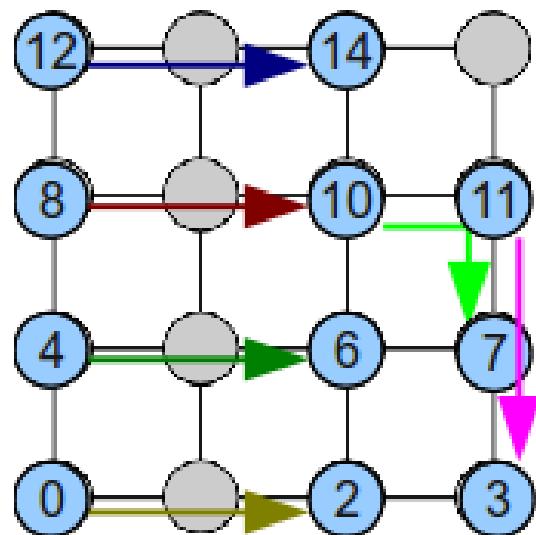
# Resultados



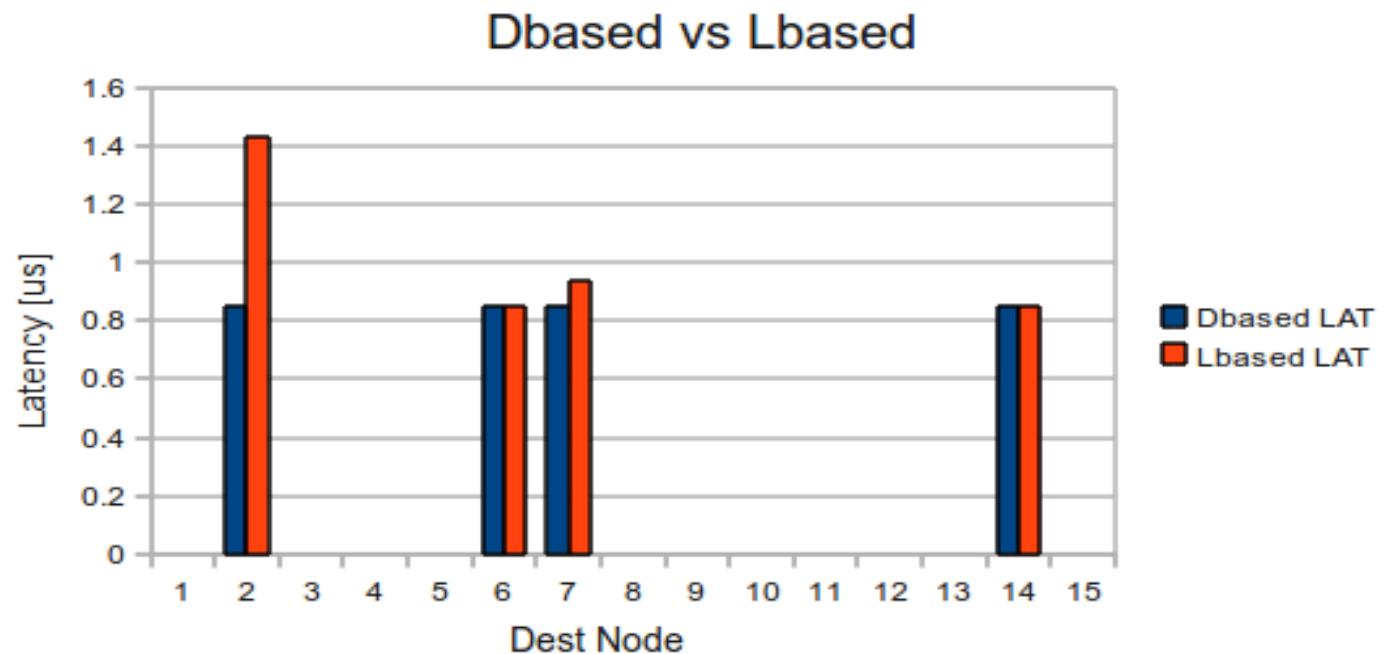
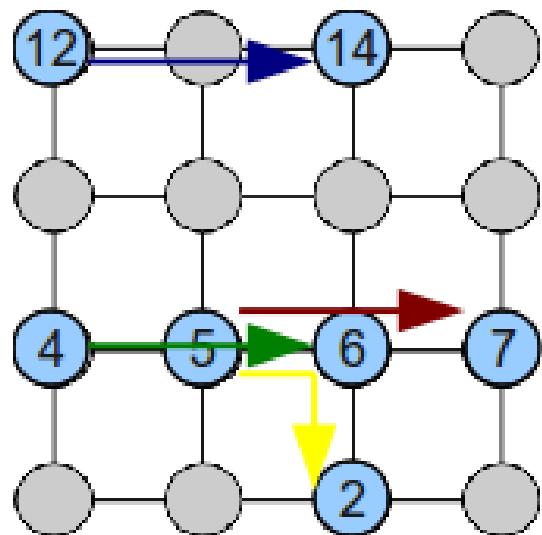
# Resultados



# Resultados

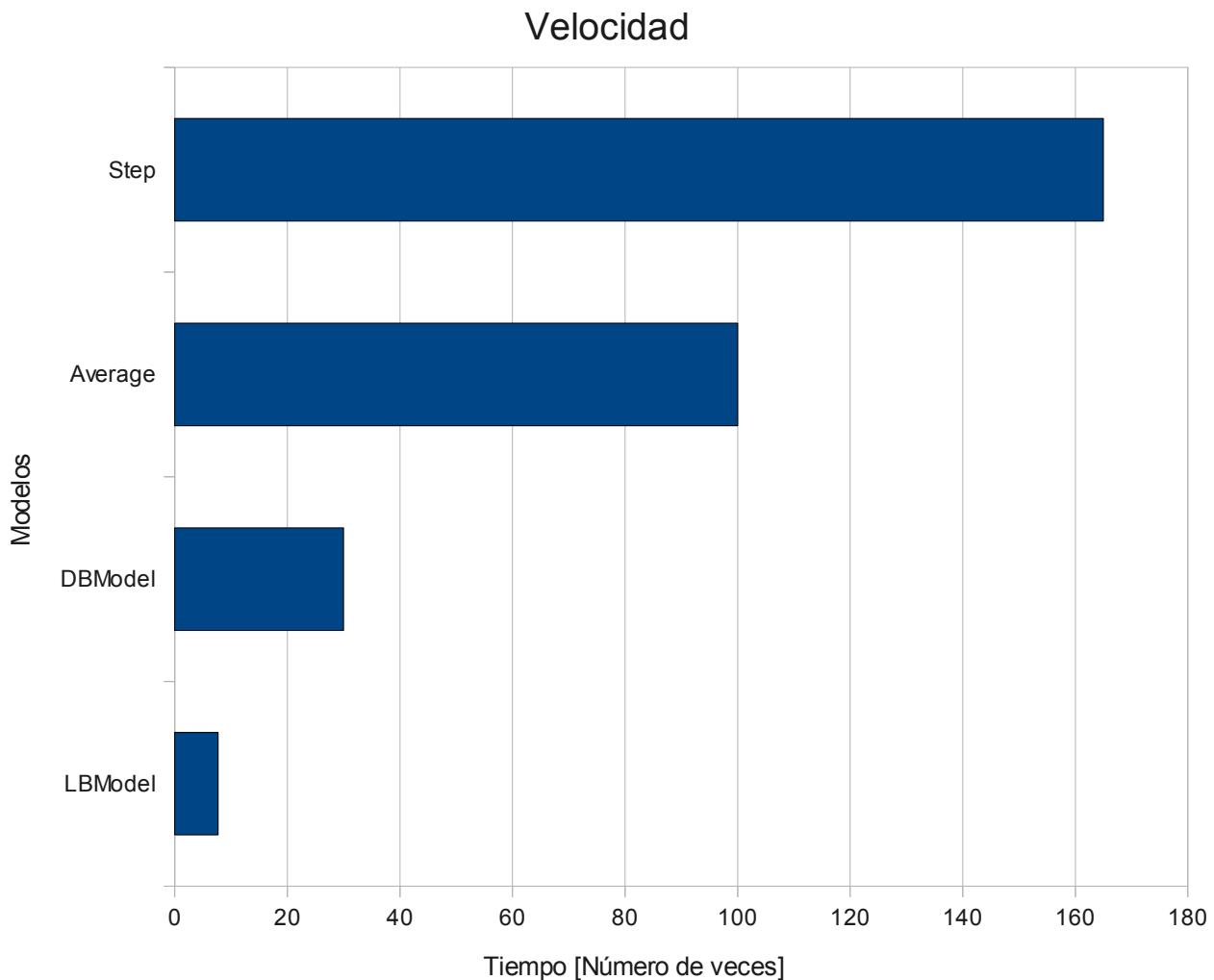


# Resultados



# Resultados

- Con una traza de 700.000 paquetes, DBModel tarda 147 seg y LBModel 574 seg.
- LBModel es 3.9 veces más lento que el DBModel, pero es más rápido que Opnet.
- LBModel detecta los choques en los links, lo que lo hace más preciso que el DBModel



# Modelo de red Lbased

Introducción
Marco Teórico
Metodología
Resultados
Conclusiones

- En este trabajo, se han estudiado algunos modelos de redes de interconexión y se ha hecho el diseño e implementación del modelo basado en enlace LBModel.
- Por lo visto, durante el estudio de los modelos existentes, Step, Average y Dbased, la velocidad y precisión son antagónicas, es decir, que si se aumenta en velocidad, se disminuye en precisión y viceversa.
- Cuando se compara con el DBModel, se ve que cuando los paquetes comparten el mismo enlace pero no el mismo destino, elLBModel hace la detección y generar una contención en el enlace
- Se cumple el compromiso entre velocidad y precisión, ya que el LBModel es más preciso qu el DBModel y más rápido que Opnet.

# Gracias!