

PPComp

Mestrado Profissional em
Computação Aplicada



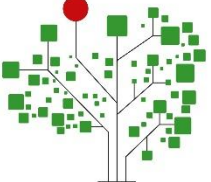
INSTITUTO FEDERAL
Espírito Santo
Campus Serra

MTS-PolKA: Divisão de tráfego multicaminho em proporção de peso com roteamento na fonte

Giancarlo O. dos Santos¹, Cristina K. Dominicini¹, Gilmar L. Vassoler¹, Rafael S. Guimarães¹, Isis Oliveira¹, Domingos Jose P. Paraíso¹ e Rodolfo S. Villaca²

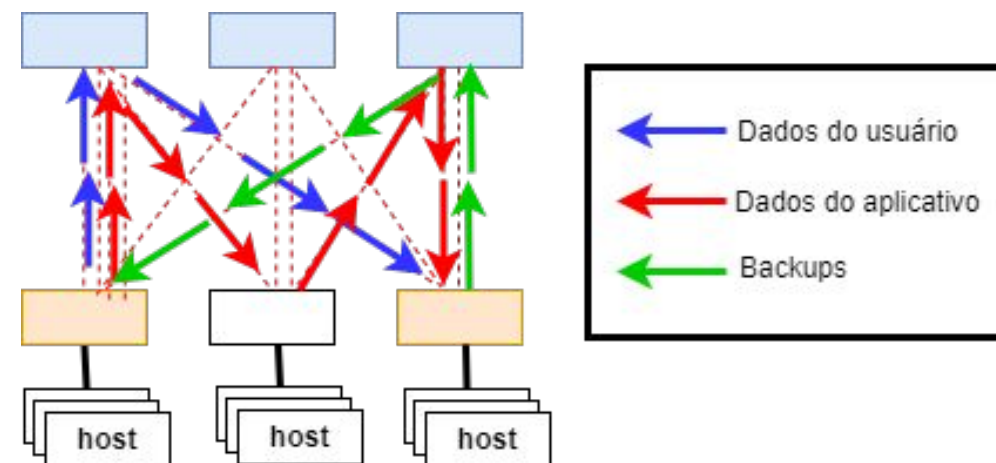
¹Programa de Pós-graduação em Computação Aplicada (PPComp)
Campus Serra do Instituto Federal do Espírito Santo (IFES)

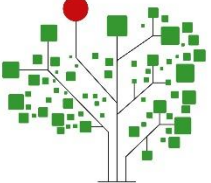
²Programa de Pós-Graduação em Informática (PPGI)
Universidade Federal do Espírito Santo (Ufes)



Motivação

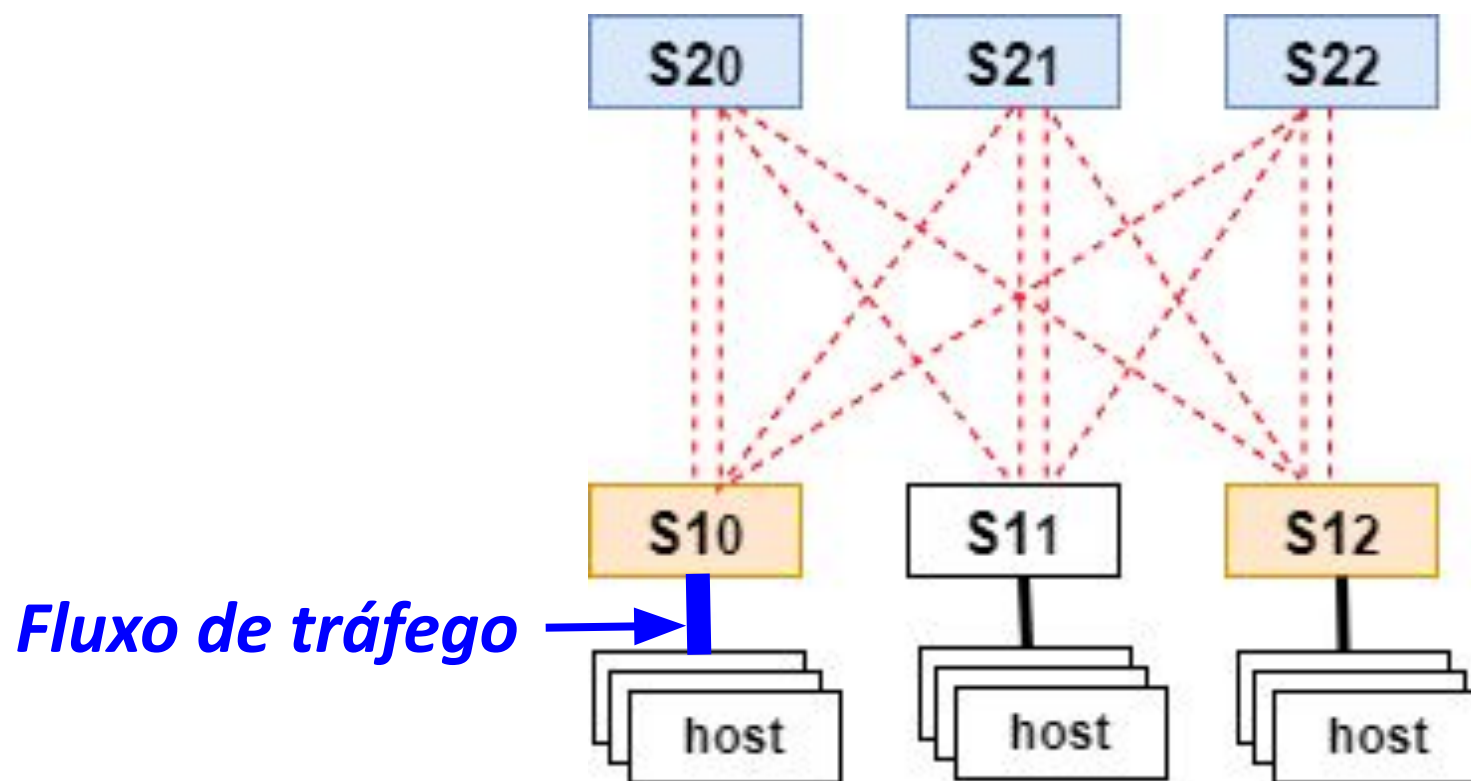
- Datacenters modernos são fundamentais para a infraestrutura de computação da Internet e processam grandes volumes de dados de forma eficiente.
- Escalabilidade é crucial para suportar o crescimento contínuo dos dados.
- Topologias como Fat-Tree e Clos facilitam a transferência eficiente de grandes volumes de dados.
- Balanceamento de carga otimiza a utilização da largura de banda disponível.

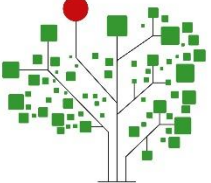




Problema

- Balancear o tráfego de forma proporcional e ágil, considerando as variações de carga e condições da rede

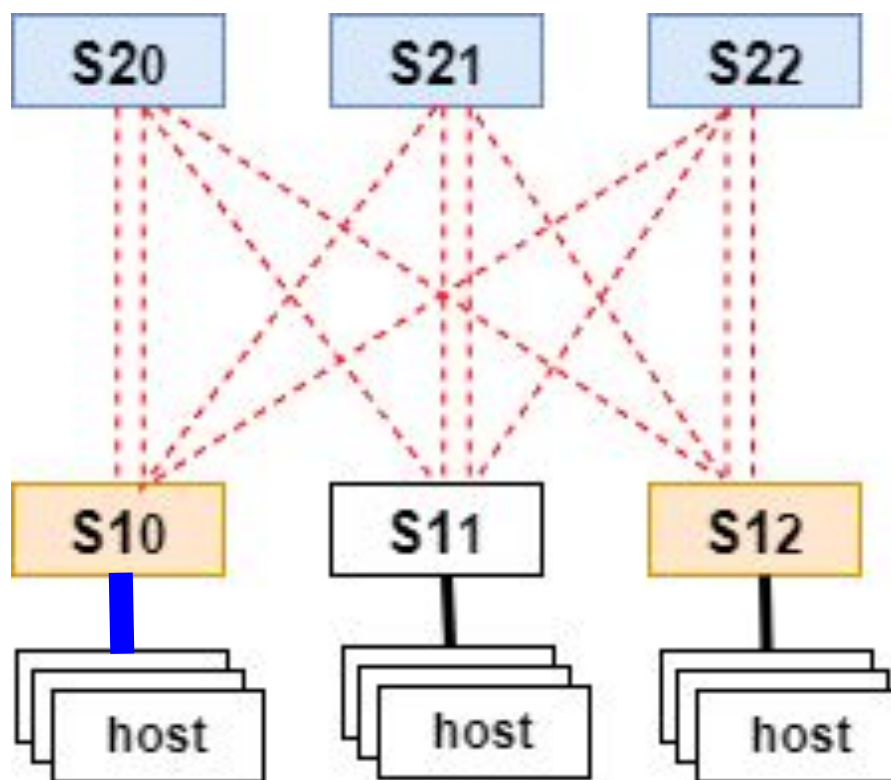


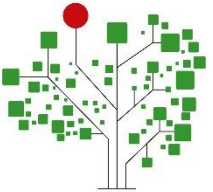


Problema

- Balancear o tráfego de forma proporcional e ágil, considerando as variações de carga e condições da rede

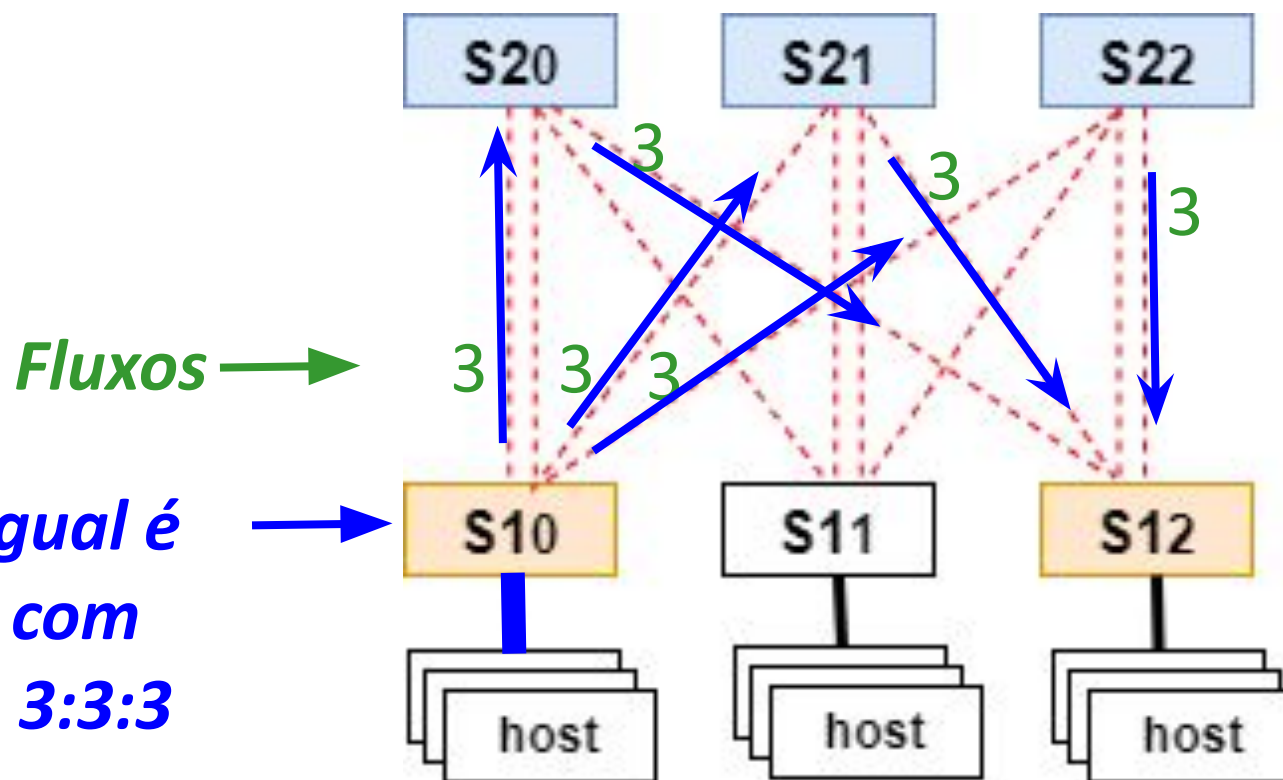
*A divisão igual é
desejada*



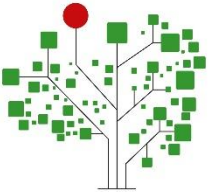


Problema

- Balancear o tráfego de forma proporcional e ágil, considerando as variações de carga e condições da rede



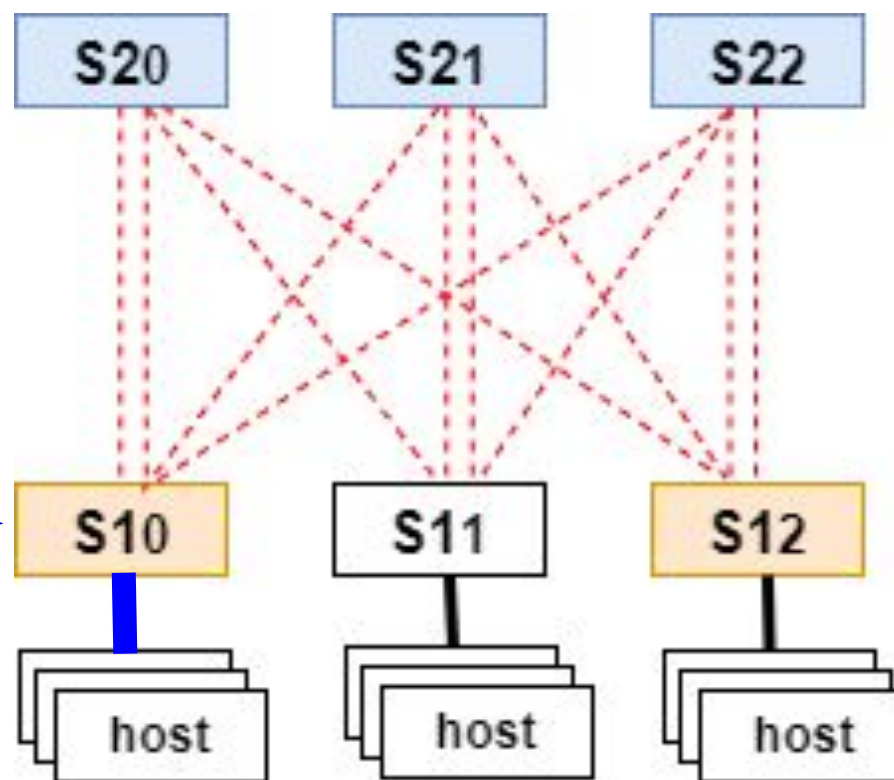
*A divisão igual é
desejada com
proporção 3:3:3*

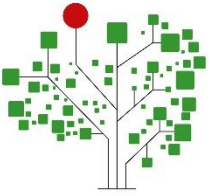


Problema

- Balancear o tráfego de forma proporcional e ágil, considerando as variações de carga e condições da rede

*A divisão desigual
é desejada*

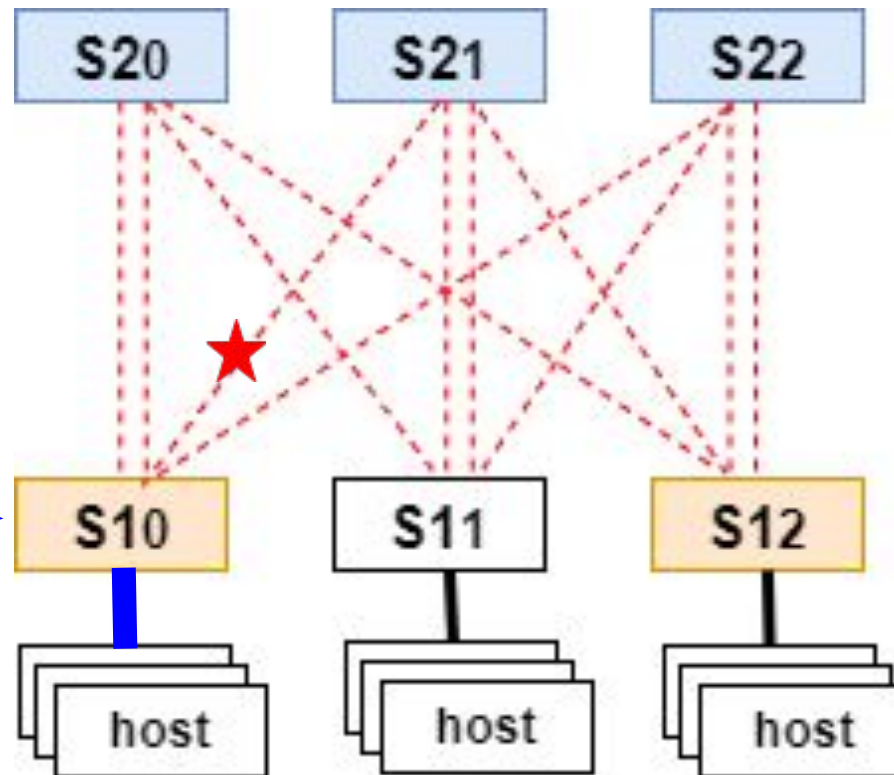


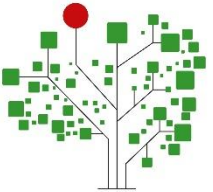


Problema

- Balancear o tráfego de forma proporcional e ágil, considerando as variações de carga e condições da rede

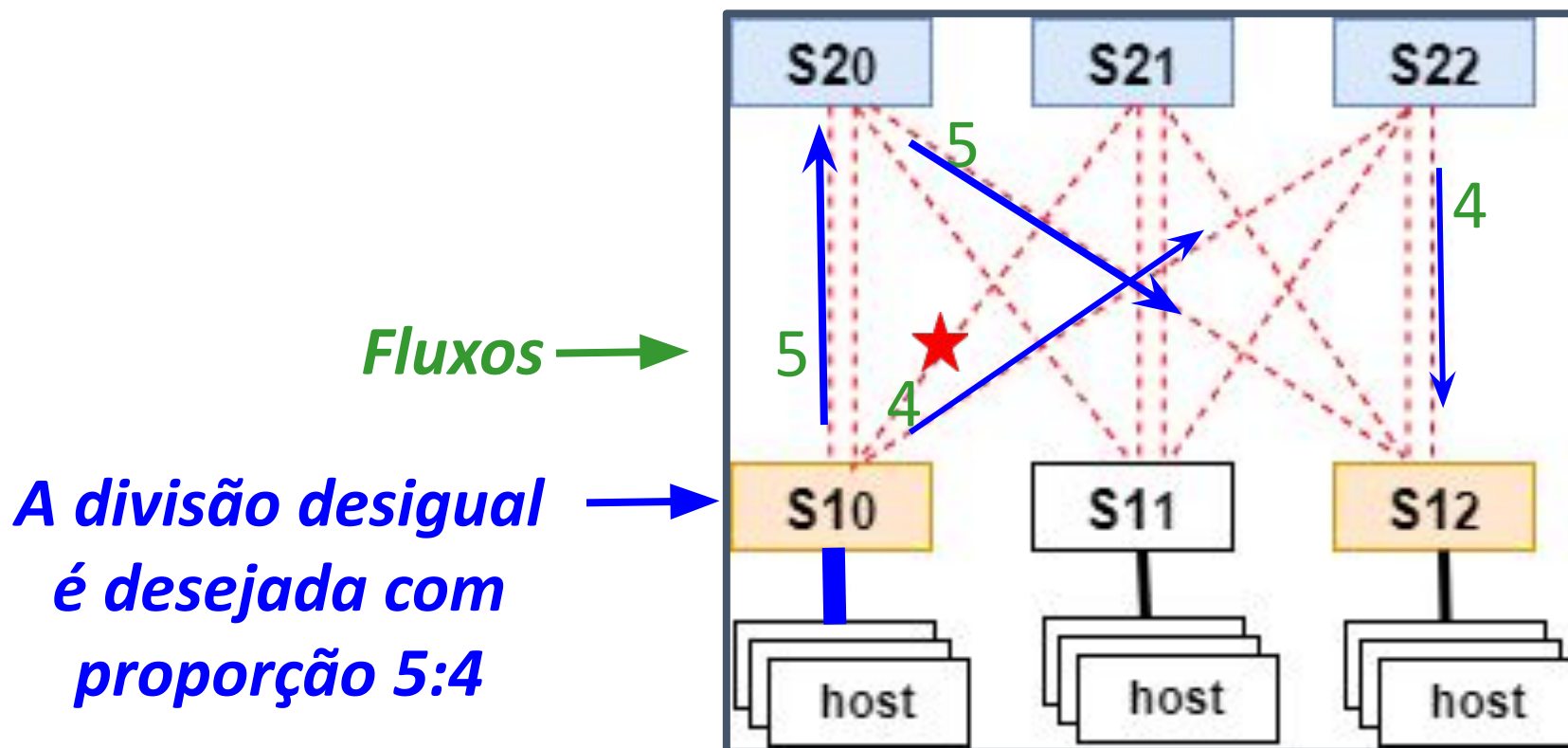
*A divisão desigual
é desejada*

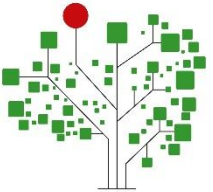




Problema

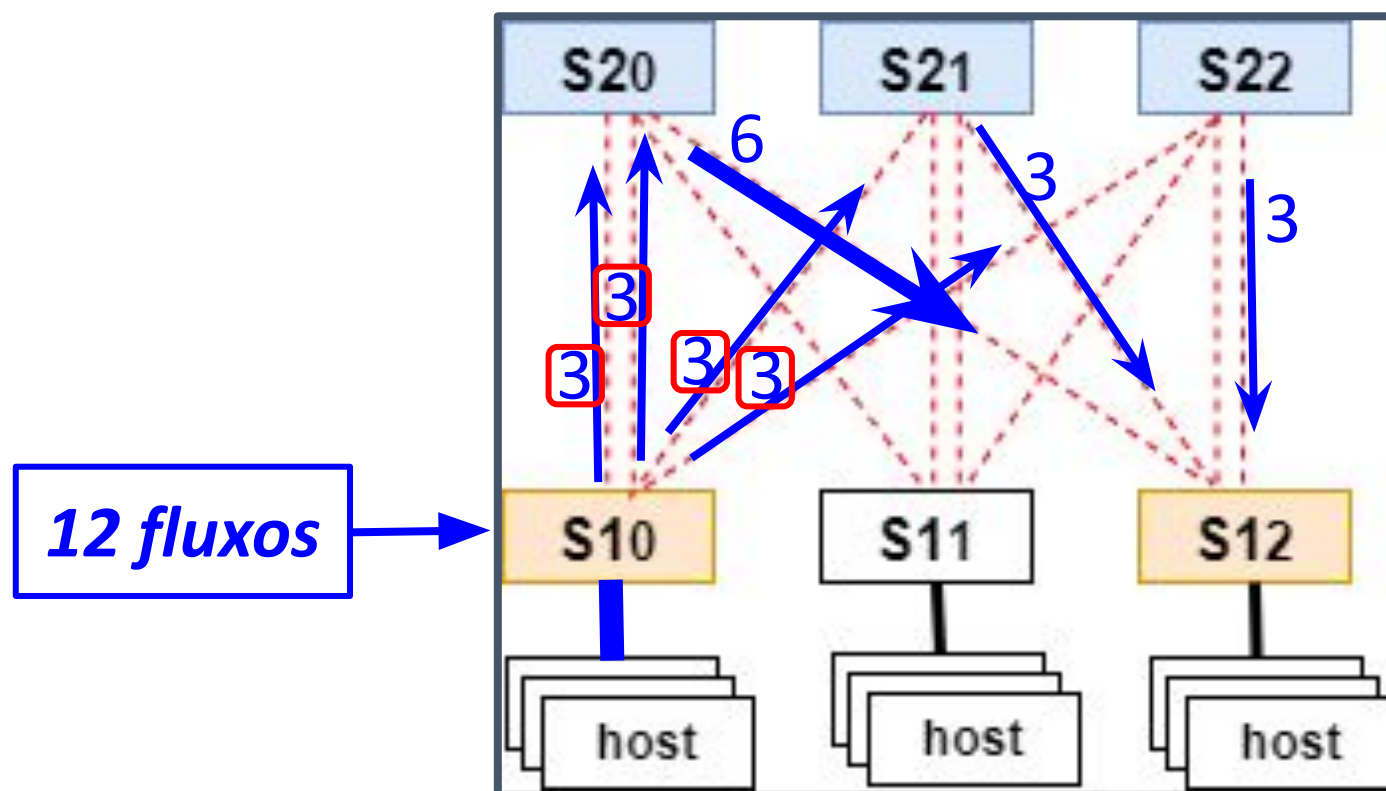
- Balancear o tráfego de forma proporcional e ágil, considerando as variações de carga e condições da rede

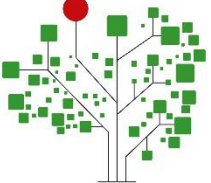




Trabalhos relacionados

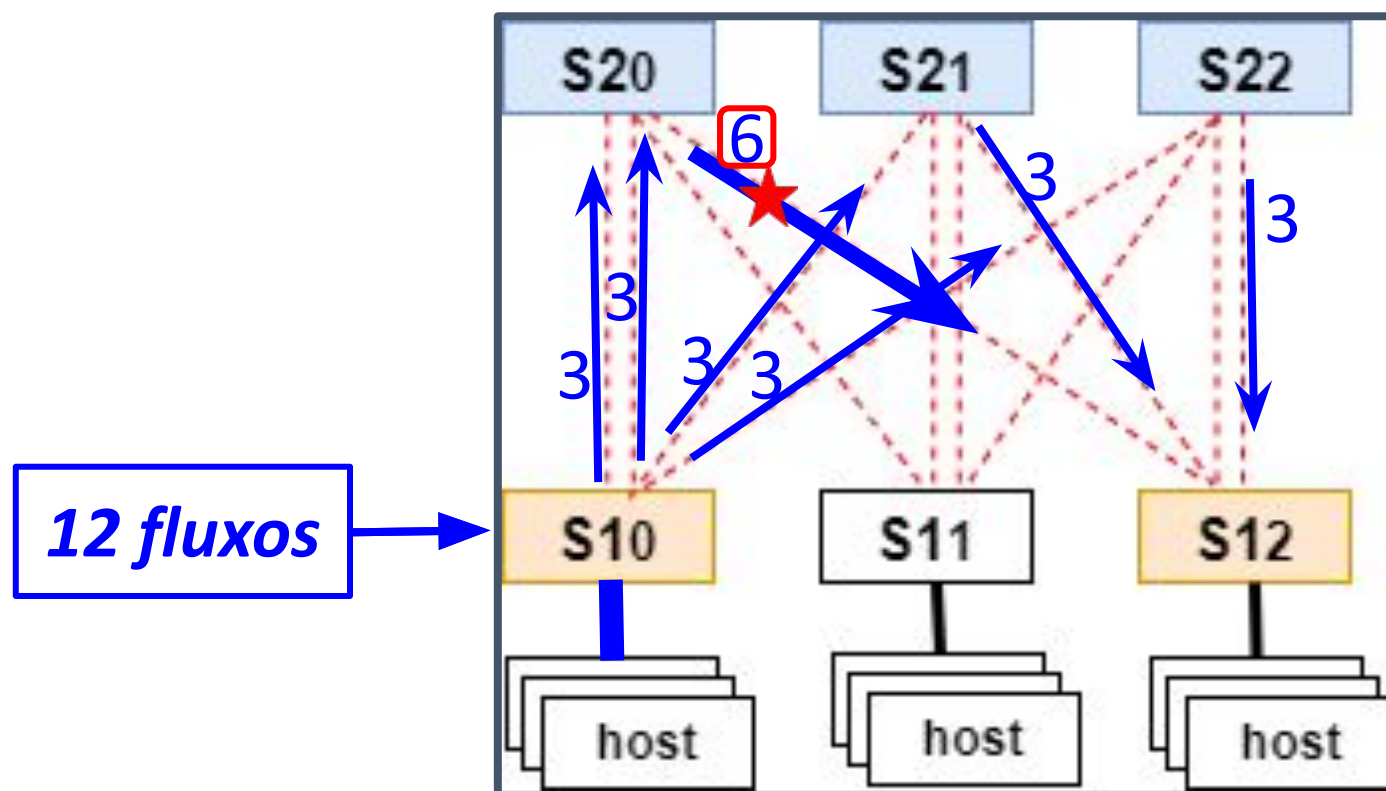
- Equal Cost Multipath - (ECMP)

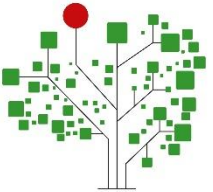




Trabalhos relacionados

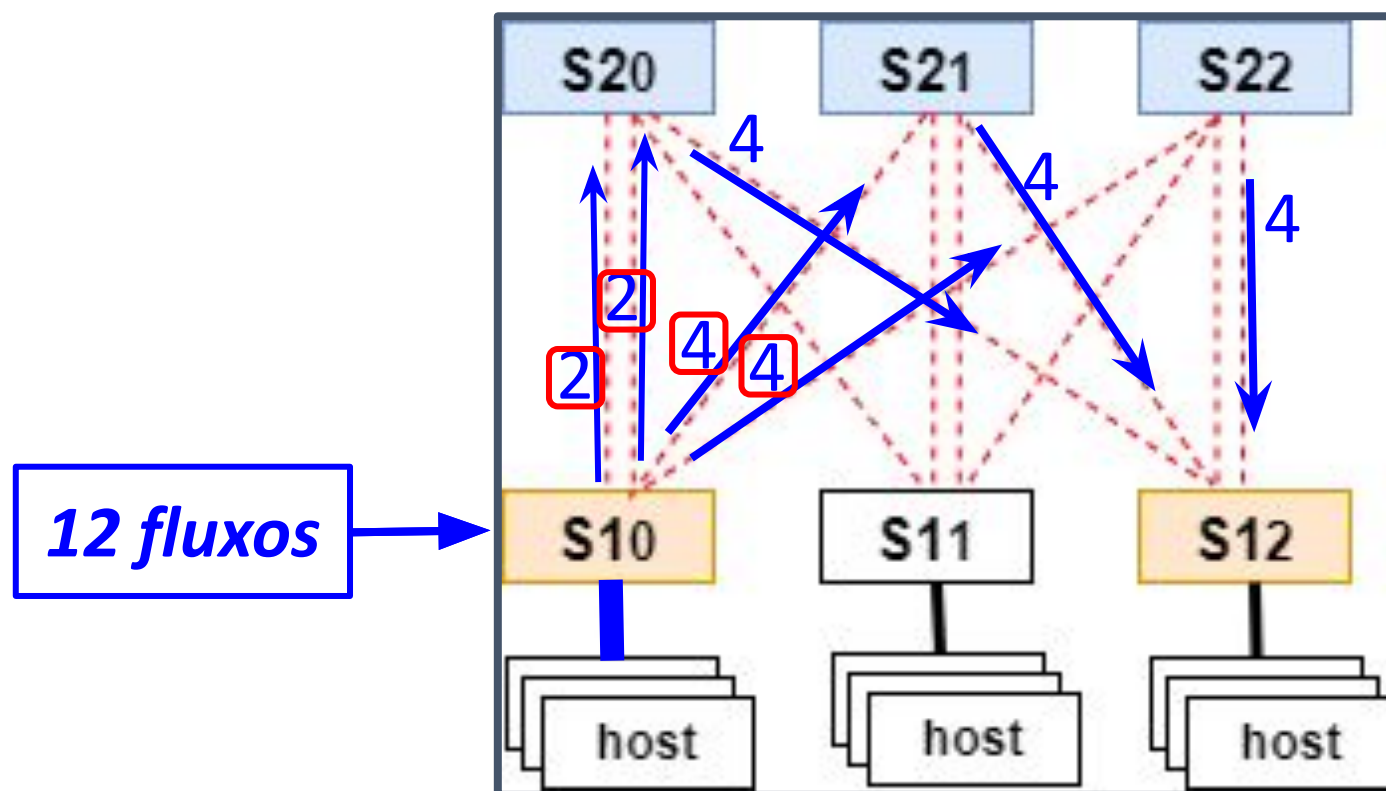
- Equal Cost Multipath - (ECMP)

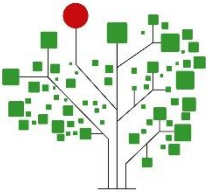




Trabalhos relacionados

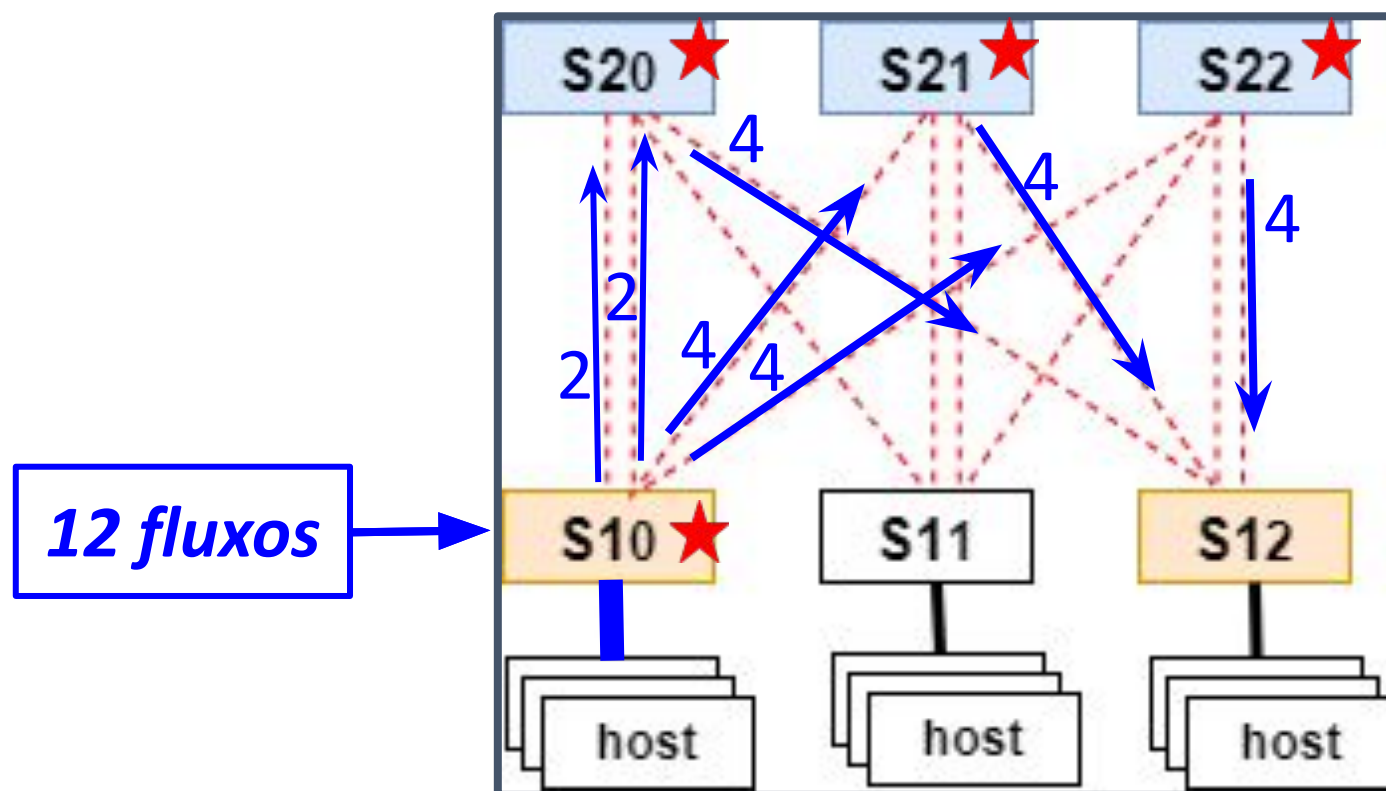
- Weighted-Cost Multipath - (WCMP)

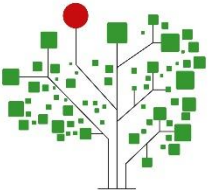




Trabalhos relacionados

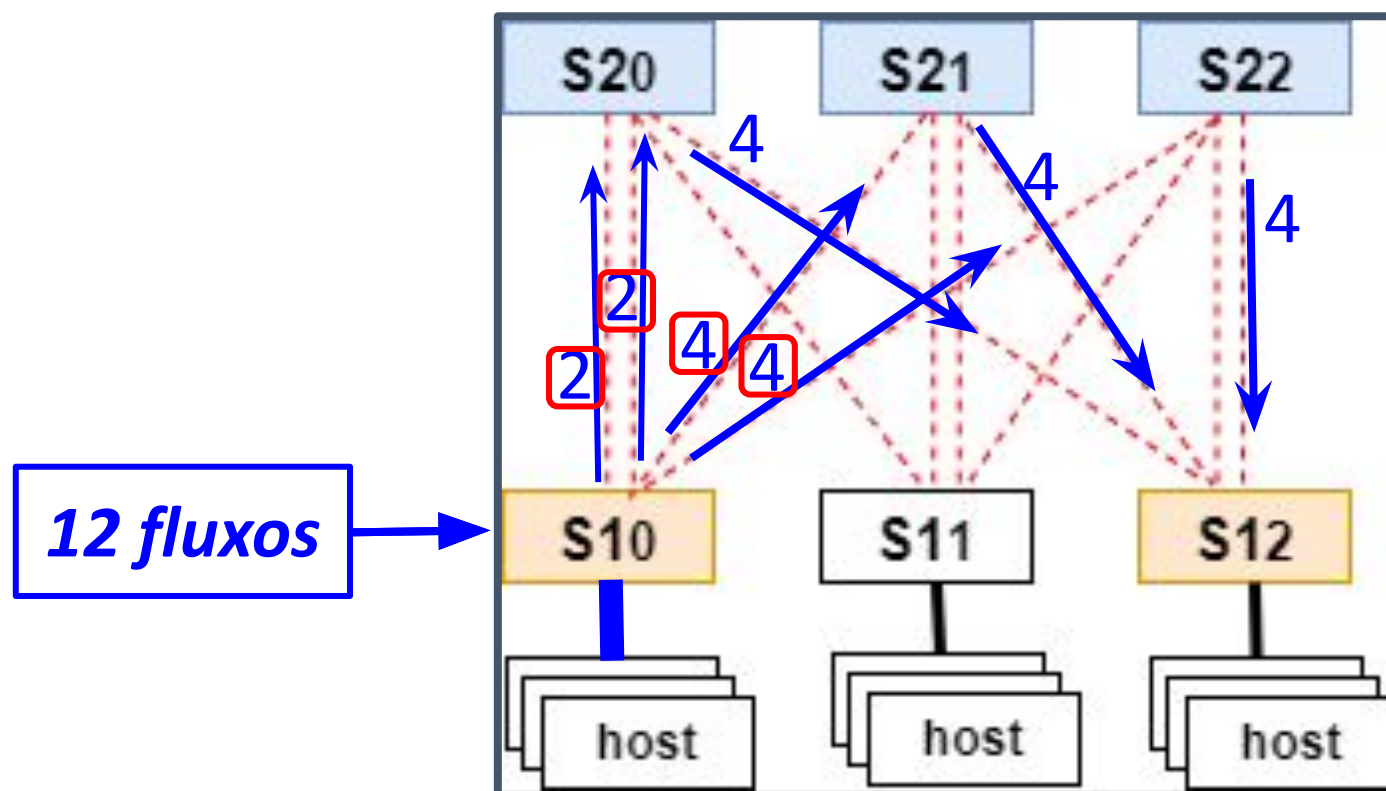
- Weighted-Cost Multipath - (WCMP)

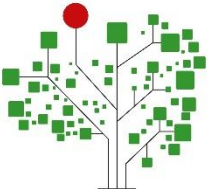




Objetivo

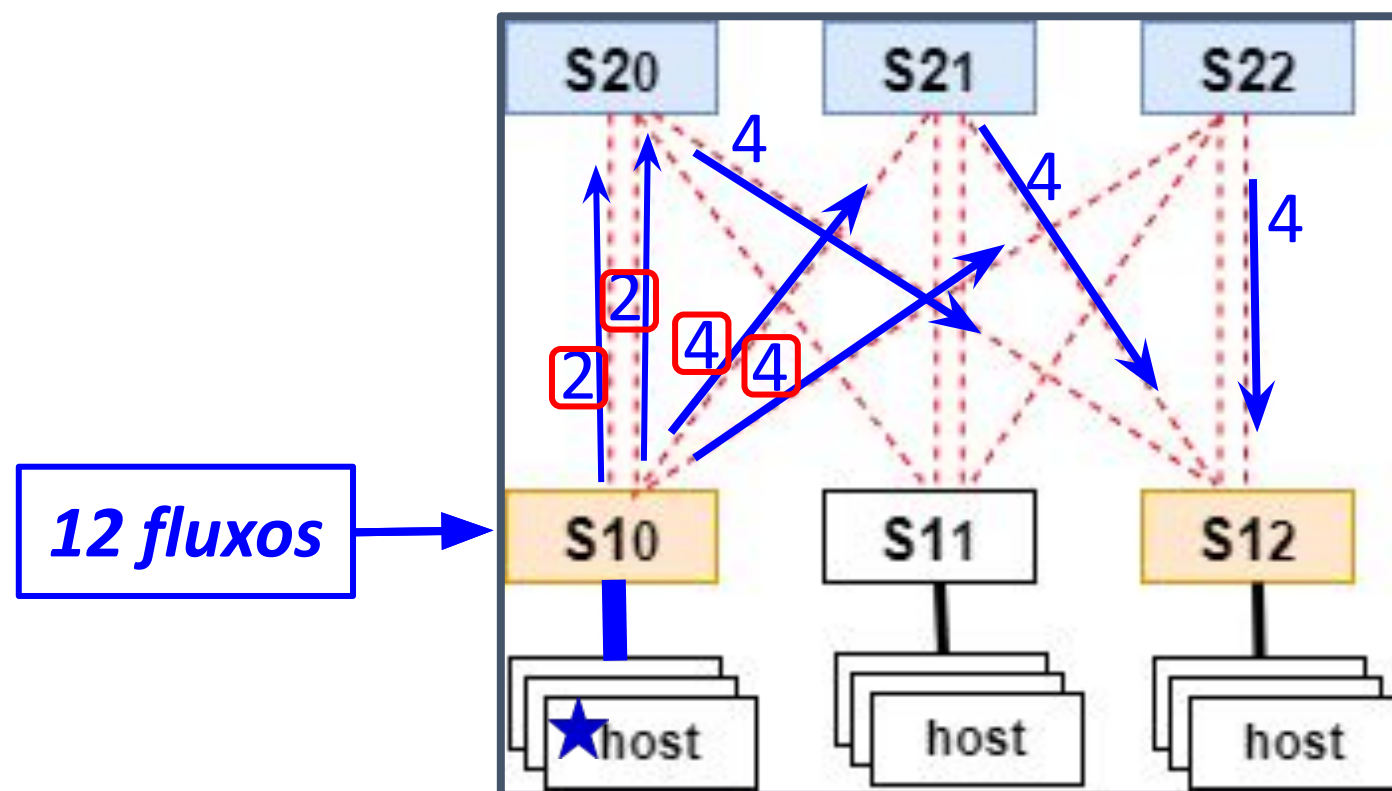
- MTS-Polka





Objetivo

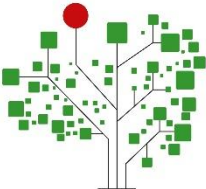
- MTS-Polka



M-Polka

- Arquitetura baseada em chave polinomial multicaminho para roteamento na fonte.
- Três polinômios:
 - **routeID**: um identificador de rota calculado usando o Teorema do Resto Chinês (CRT).
 - **nodeID**: para identificar cada nó principal.
 - Polinômio irredutível
 - **portID**: para identificar as portas de cada nó principal.
- O encaminhamento usa uma operação mod (resto da divisão):

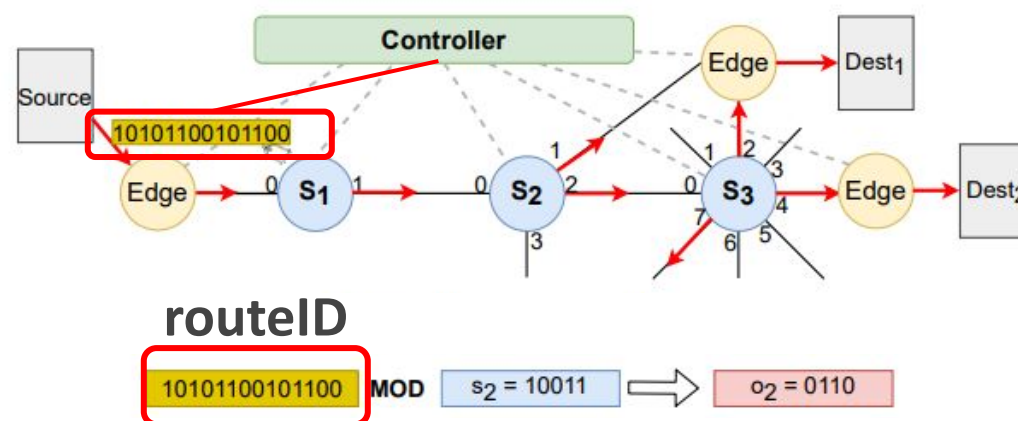
$$\text{portID} = \langle \text{routeID} \rangle_{\text{nodeID}}$$

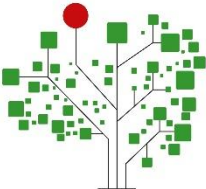


M-Polka

- Descobrimos o estado de encaminhamento usando a operação mod:
 - **routeID**

<10101100101100>





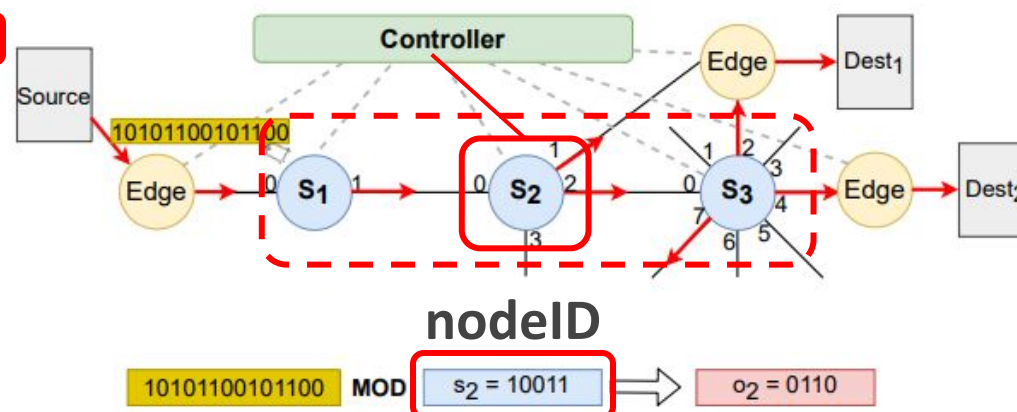
M-Polka

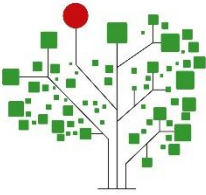
- Descobrimo o estado de encaminhamento usando a operação mod:

- **nodeID**

- $\langle 10101100101100 \rangle$

10011

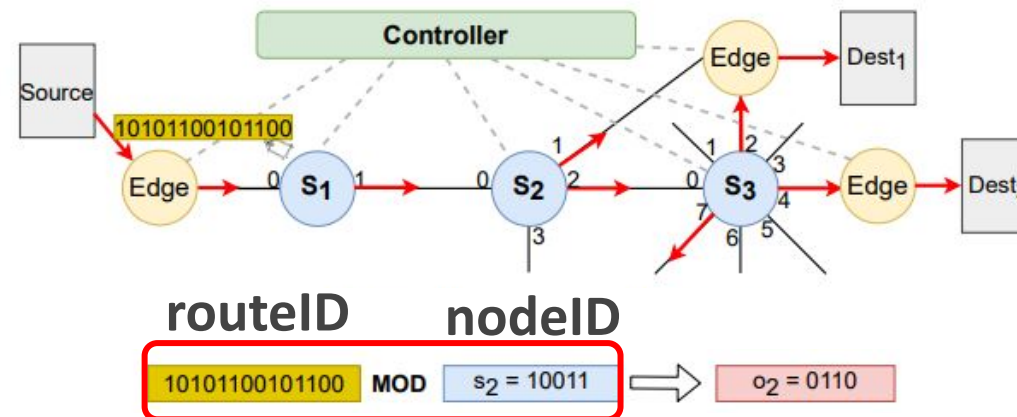


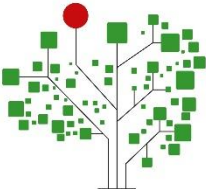


M-Polka

- Descobrimo o estado de encaminhamento usando a operação mod:

<10101100101100>



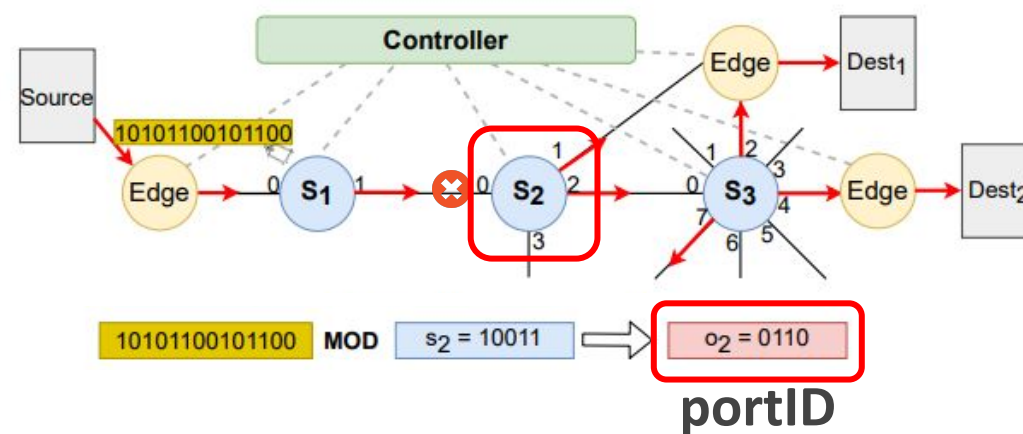


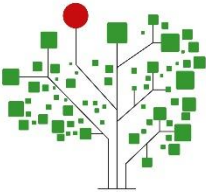
M-Polka

- Descobrimo o estado de encaminhamento usando a operação mod:

- **portID**

$\langle 10101100101100 \rangle_{10011} = 0110 \rightarrow$

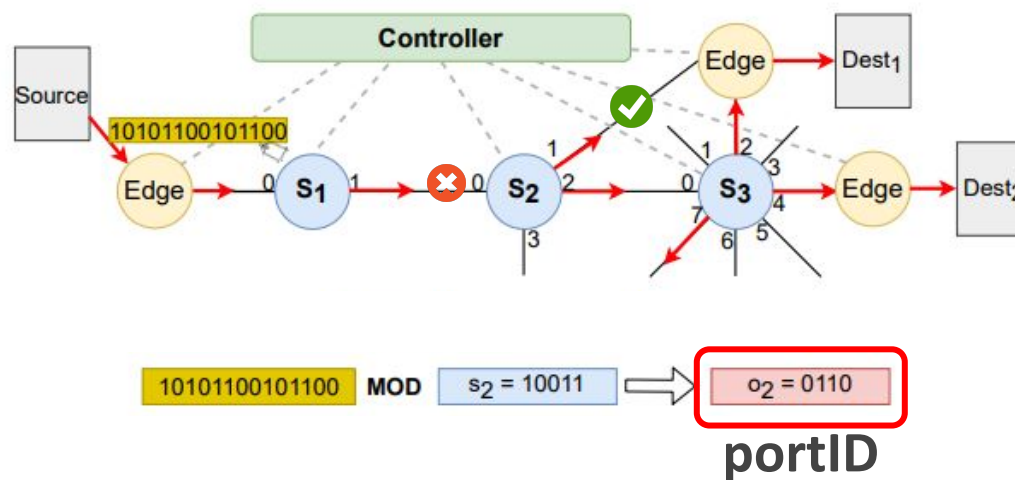


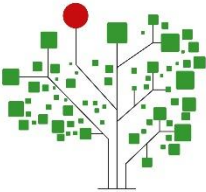


M-Polka

- Descobrimo o estado de encaminhamento usando a operação mod:

○ $\langle 10101100101100 \rangle_{10011} = 0110 \rightarrow \text{dirija para as portas 1}$

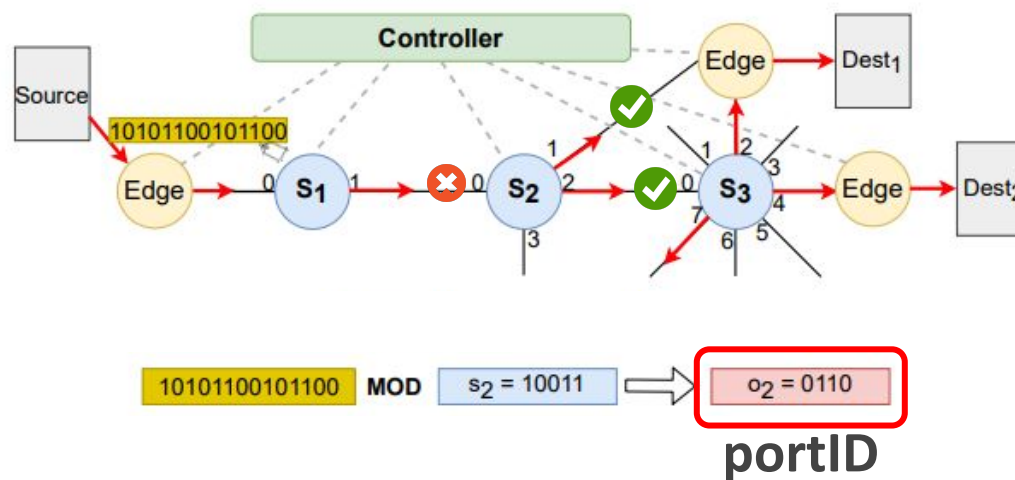


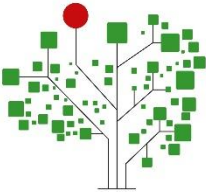


M-Polka

- Descobrimo o estado de encaminhamento usando a operação mod:

○ $\langle 10101100101100 \rangle_{10011} = 0110 \rightarrow \text{dirija para as portas 1 e 2}$

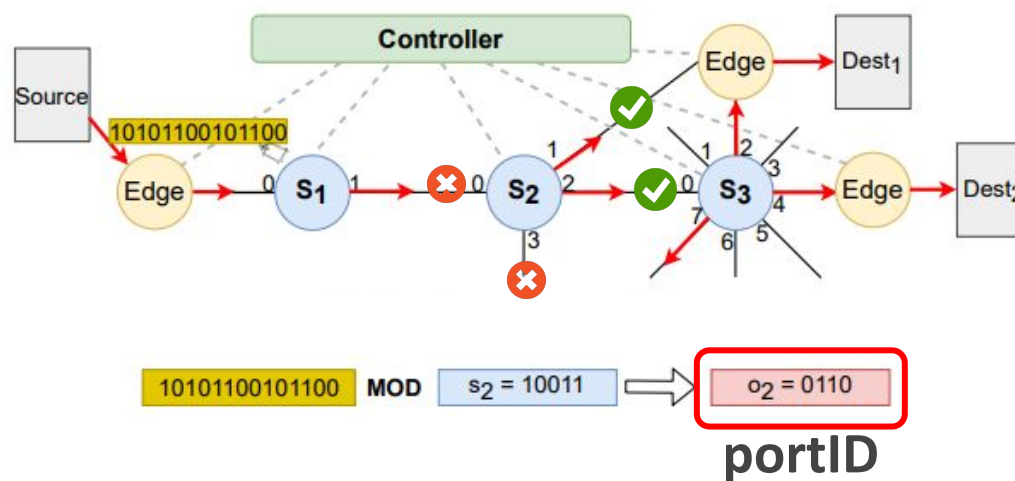


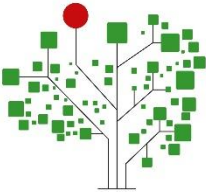


M-Polka

- Descobrimos o estado de encaminhamento usando a operação mod:

○ $\langle 10101100101100 \rangle_{10011} = 0110 \rightarrow \text{dirija para as portas 1 e 2}$

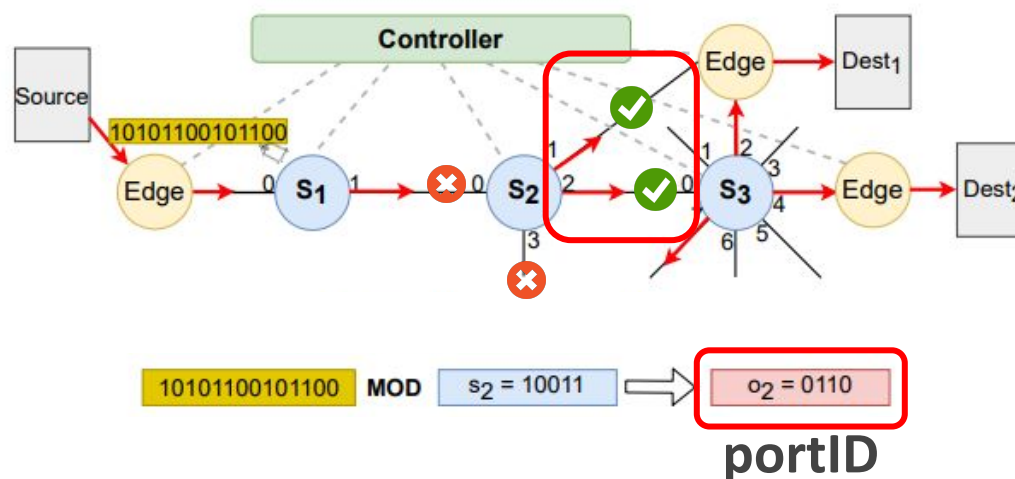




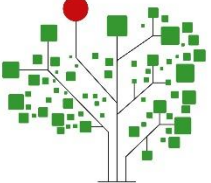
M-Polka

- Descobrimos o estado de encaminhamento usando a operação mod:

○ $\langle 10101100101100 \rangle_{10011} = 0110 \rightarrow \text{dirija para as portas 1 e 2}$



Obs.: O M-Polka clona os pacotes e não divide como é a nossa ideia.



PPComp

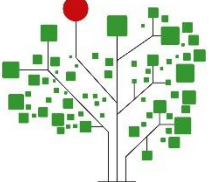
Mestrado Profissional em
Computação Aplicada



INSTITUTO FEDERAL
Espírito Santo
Campus Serra

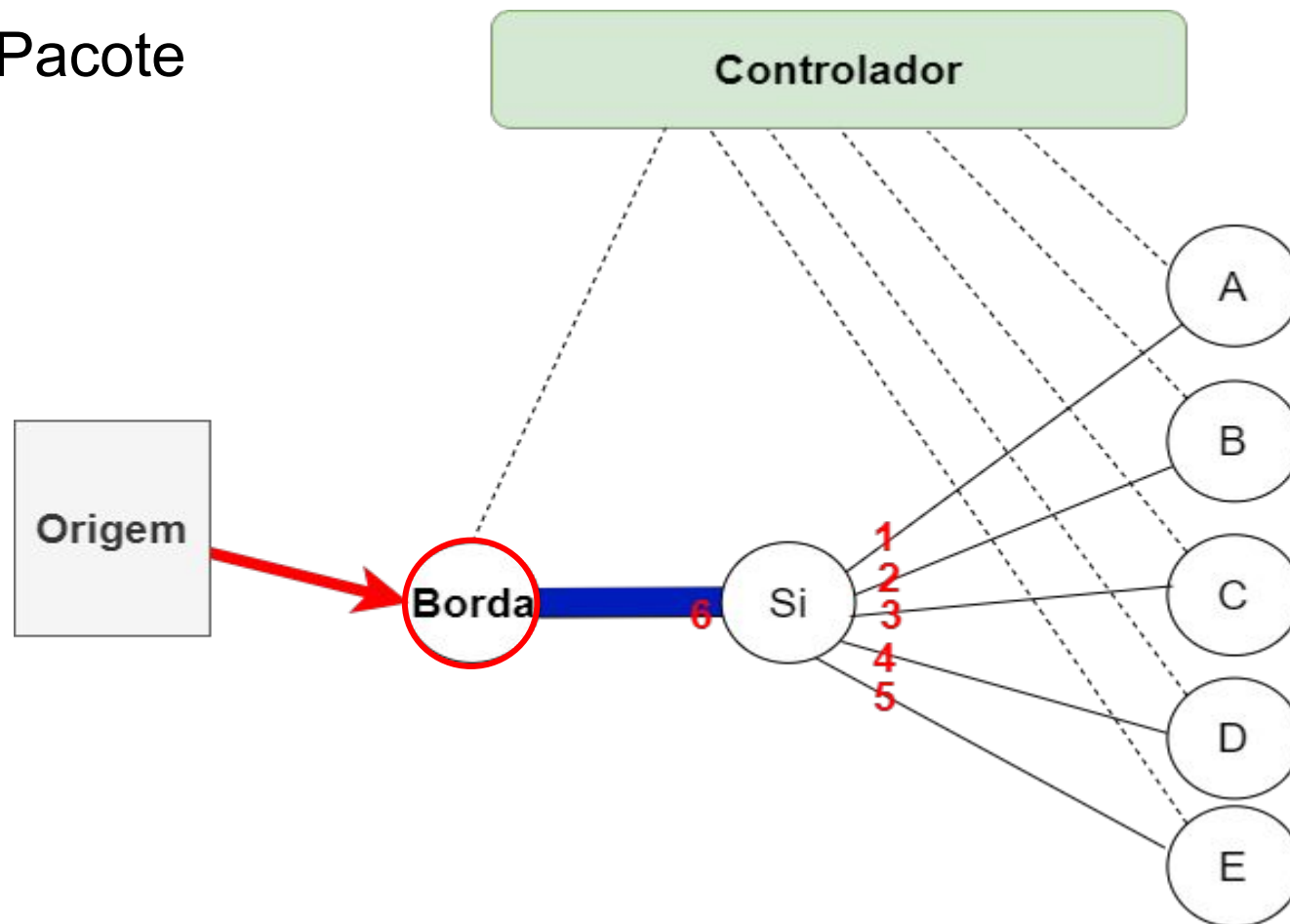
Como funciona o MTS-PolKA?

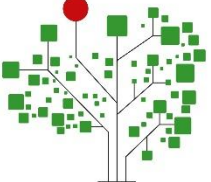
- MTS-Polka
 - Ingresso do Pacote
 - Definição de Rota
 - Seleção de Perfil
 - Distribuição de Tráfego



Como funciona o MTS-PolKA?

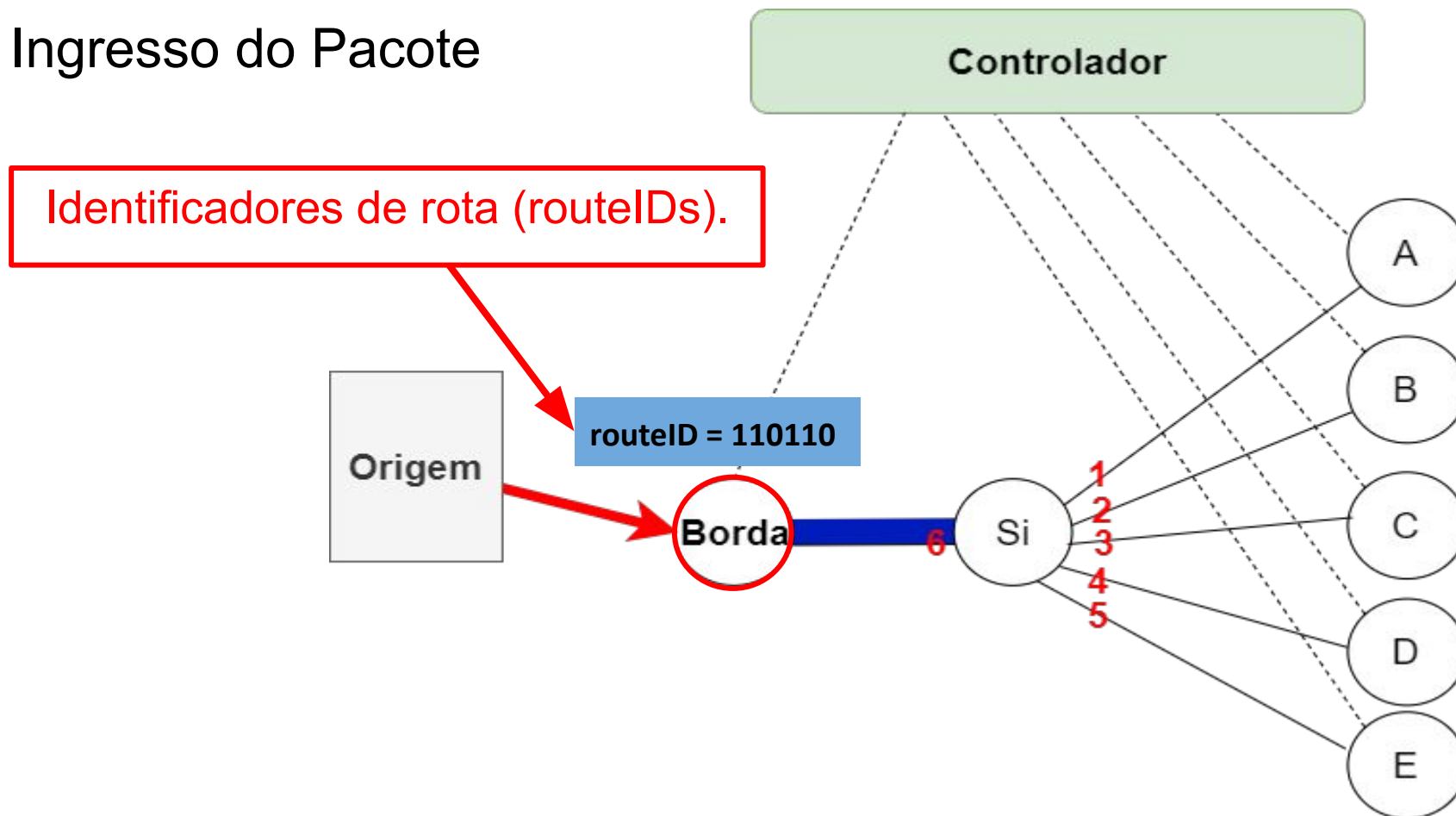
- Ingresso do Pacote

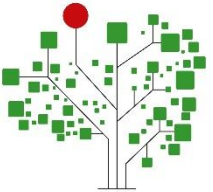




Como funciona o MTS-PolKA?

- Ingresso do Pacote

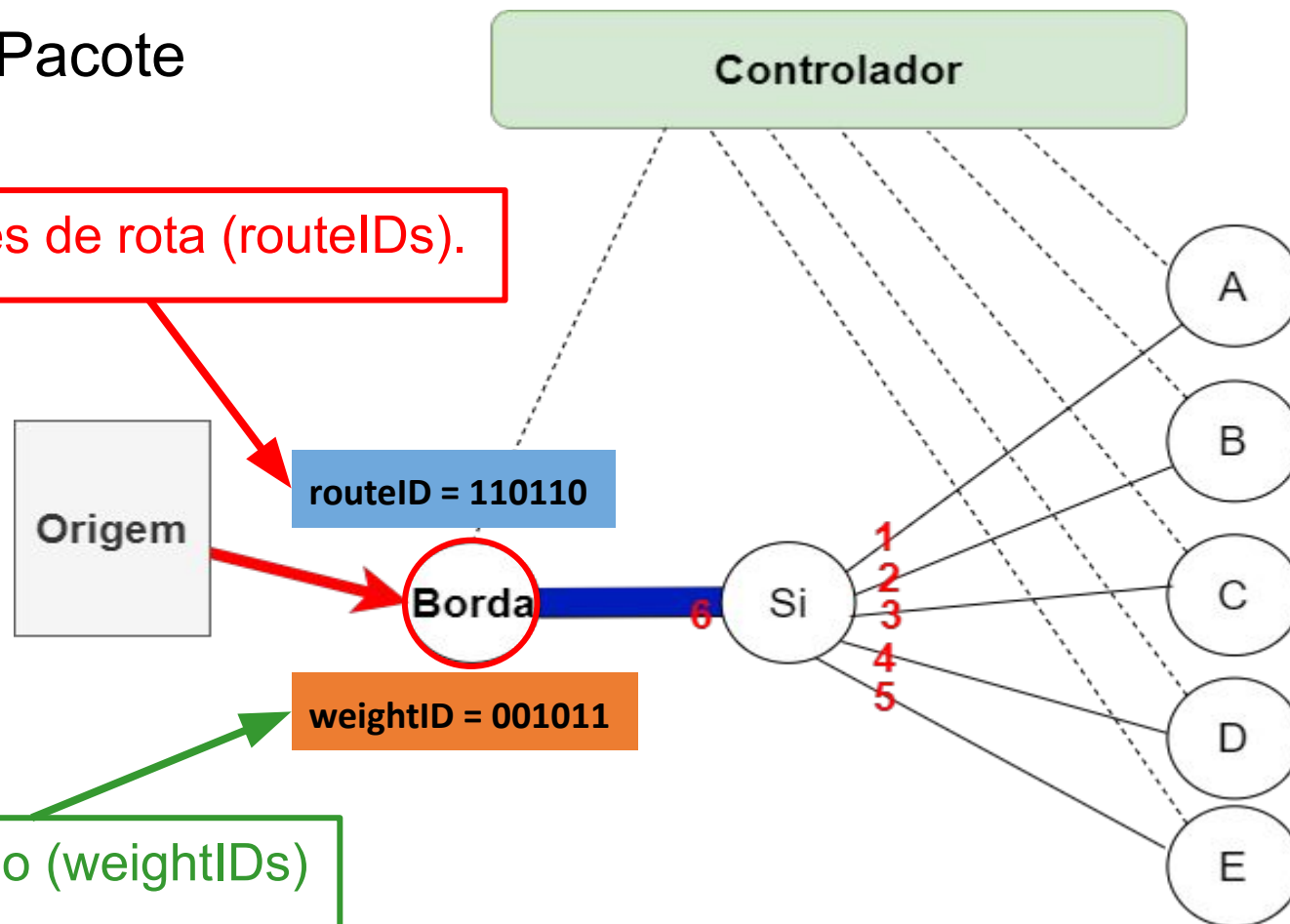




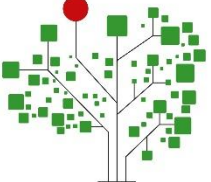
Como funciona o MTS-PolKA?

- Ingresso do Pacote

Identificadores de rota (routeIDs).



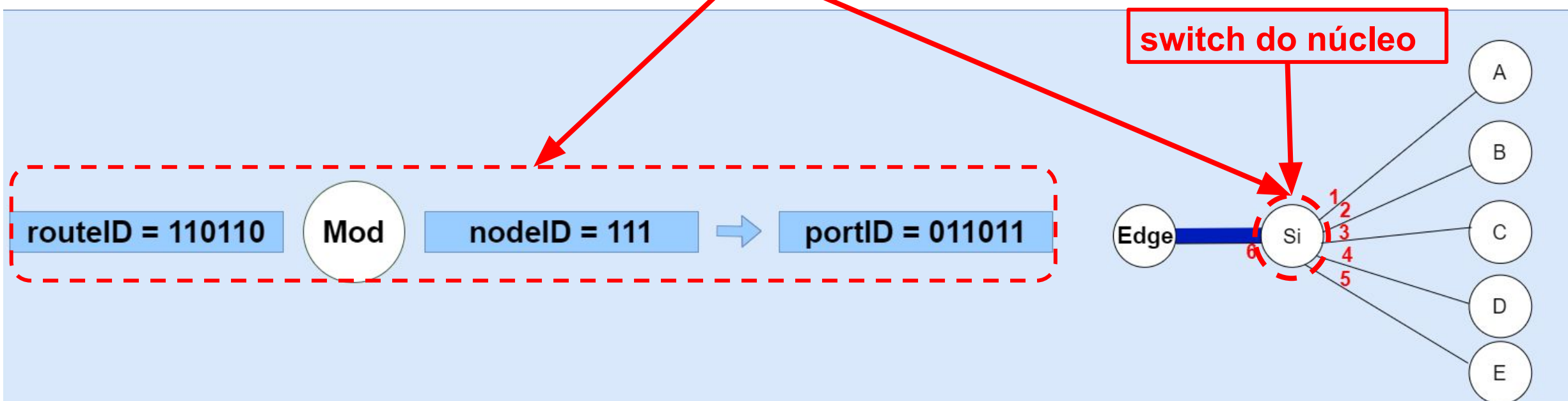
Identificadores de Peso (weightIDs)

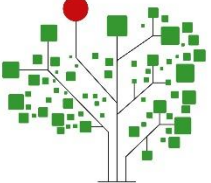


Como funciona o MTS-PolKA?

- Definição de Rota

No switch do núcleo a operação de módulo entre `routeID` e `nodeID` (111) resulta em **portID = 011011**.



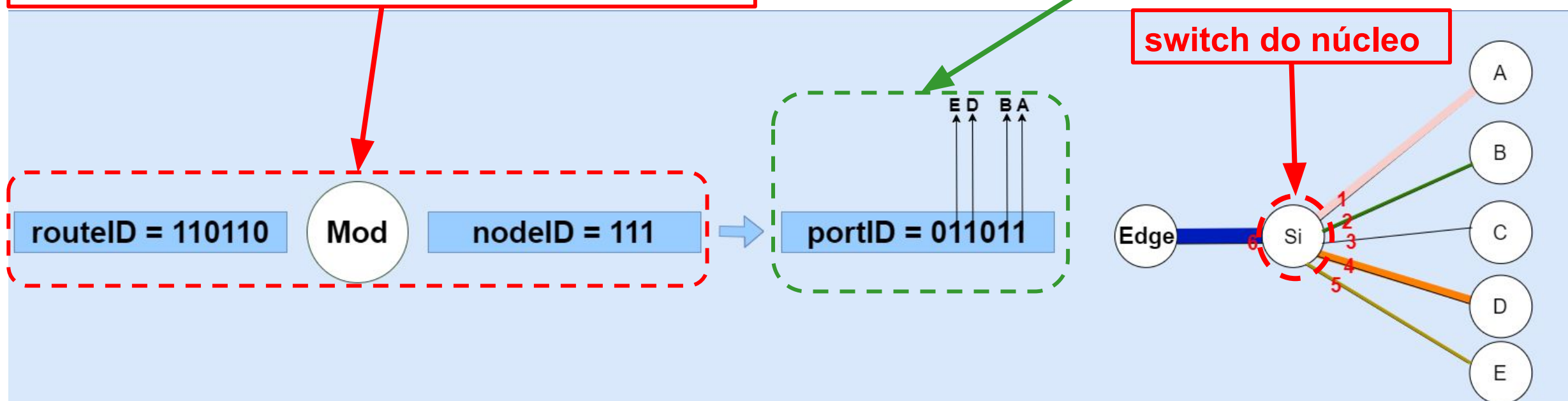


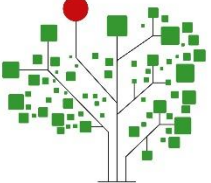
Como funciona o MTS-PolKA?

- Definição de Rota

No switch do núcleo a operação de módulo entre `routeID` e `nodeID` (111) resulta em **portID = 011011**.

Definindo as portas ativas A, B, D e E para encaminhamento

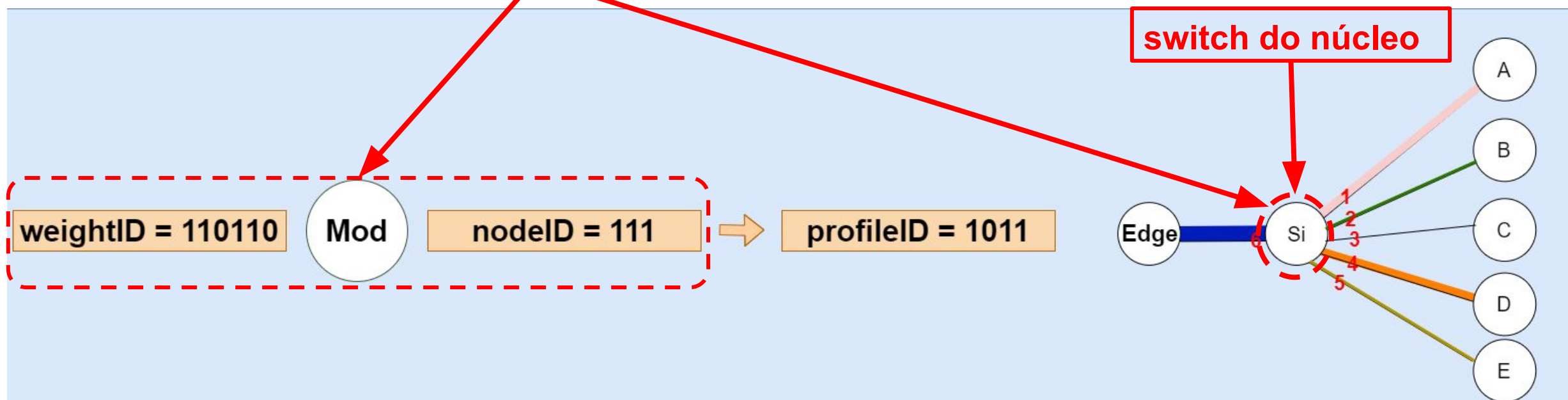


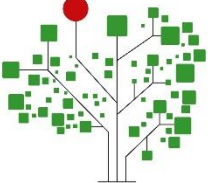


Como funciona o MTS-PolKA?

- Seleção de Perfil

No switch do núcleo a operação de módulo entre **weightID** e **nodeID** (111) resulta no **profileID = 1011** (binário).

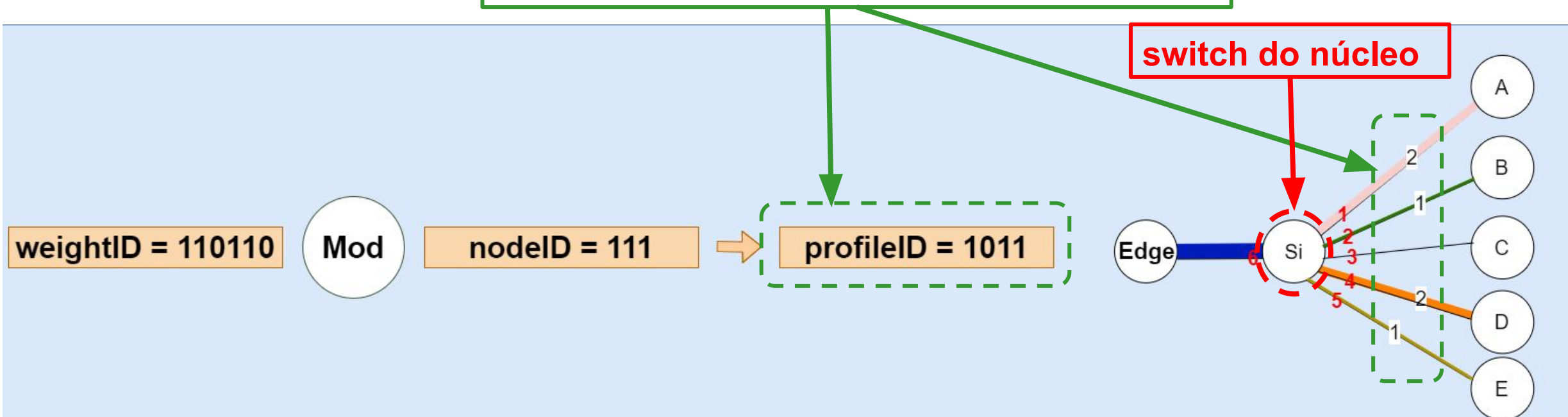


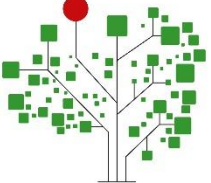


Como funciona o MTS-PolKA?

- Seleção de Perfil

Definindo o **profileID 1011** (11) a proporção **2:1:2:1** para as portas ativas conforme o **portID = 011011** (definido no cálculo da rota)





Como funciona o MTS-PolKA?

- Seleção de Perfil

Switches do núcleo mantêm tabelas estáticas com perfis de divisão de tráfego, evitando reconfigurações.

A operação de módulo entre **weightID** e **nodeID** resulta em **profileID = 1011**.

weightID = 001011

Mod

nodeID = 111

profileID = 1011

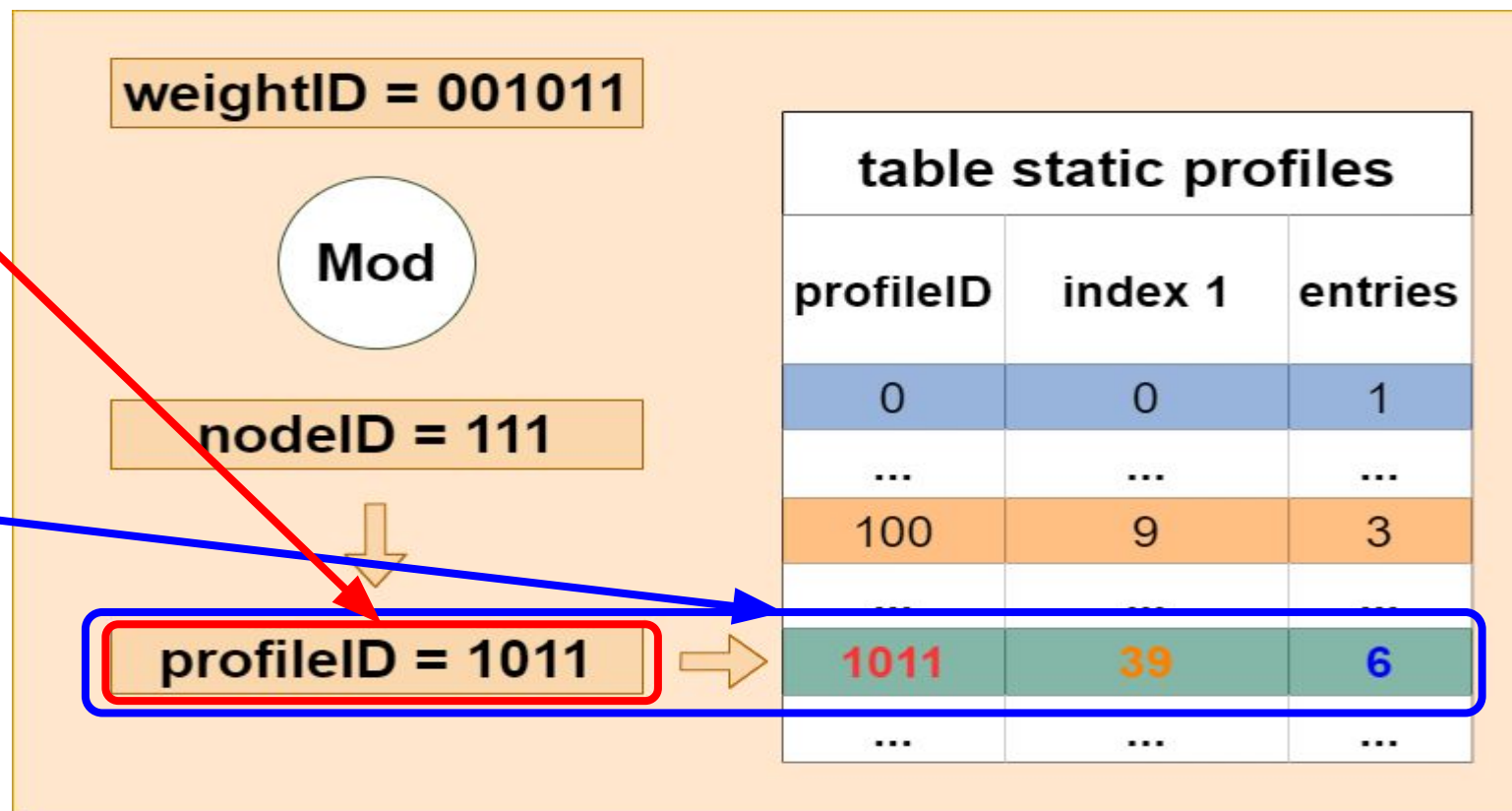
table static profiles

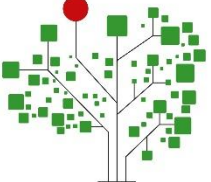
profileID	index 1	entries
0	0	1
...
100	9	3
...
1011	39	6
...

Como funciona o MTS-PolKA?

A operação de módulo entre weightID e nodeID resulta em **profileID = 1011**.

Aponta para um perfil de divisão de tráfego (profileID = 1011) na table static profiles

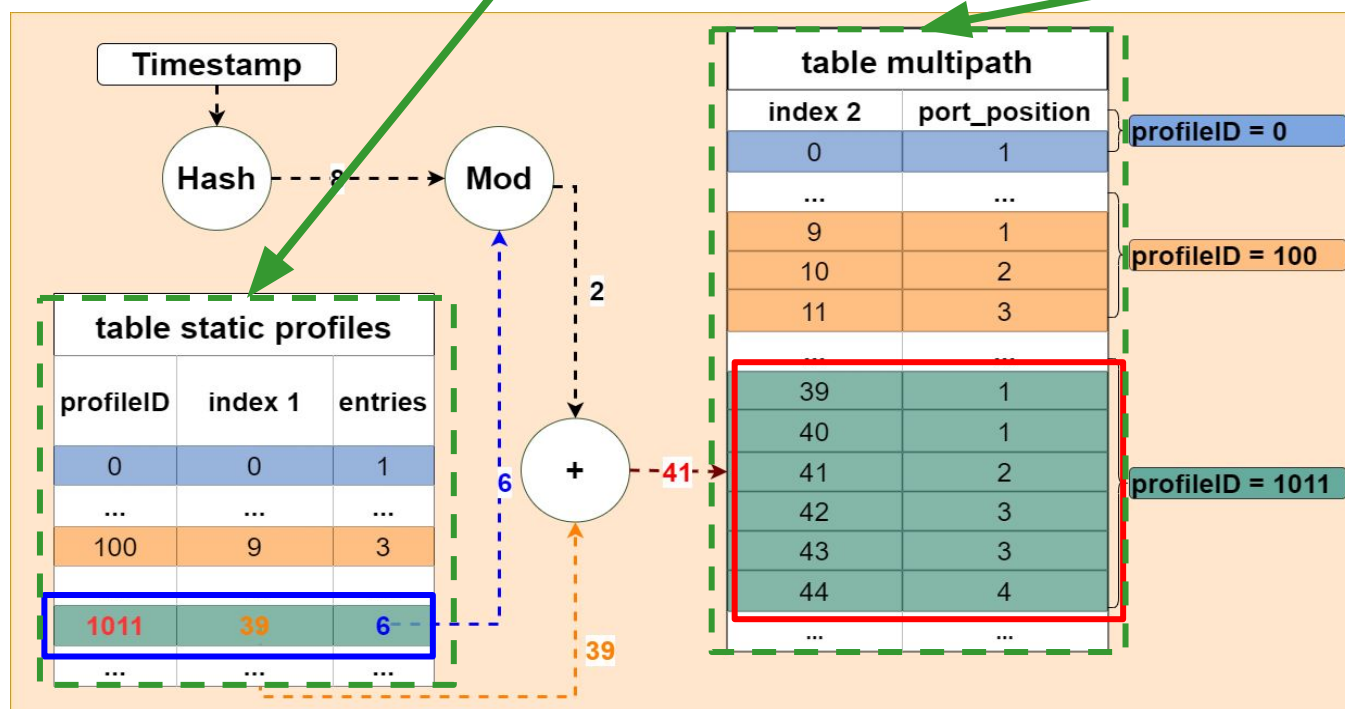


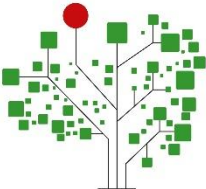


Como funciona o MTS-PolKA?

- Distribuição de Tráfego

Switches do núcleo mantêm tabelas estáticas com perfis de divisão de tráfego, evitando reconfigurações.

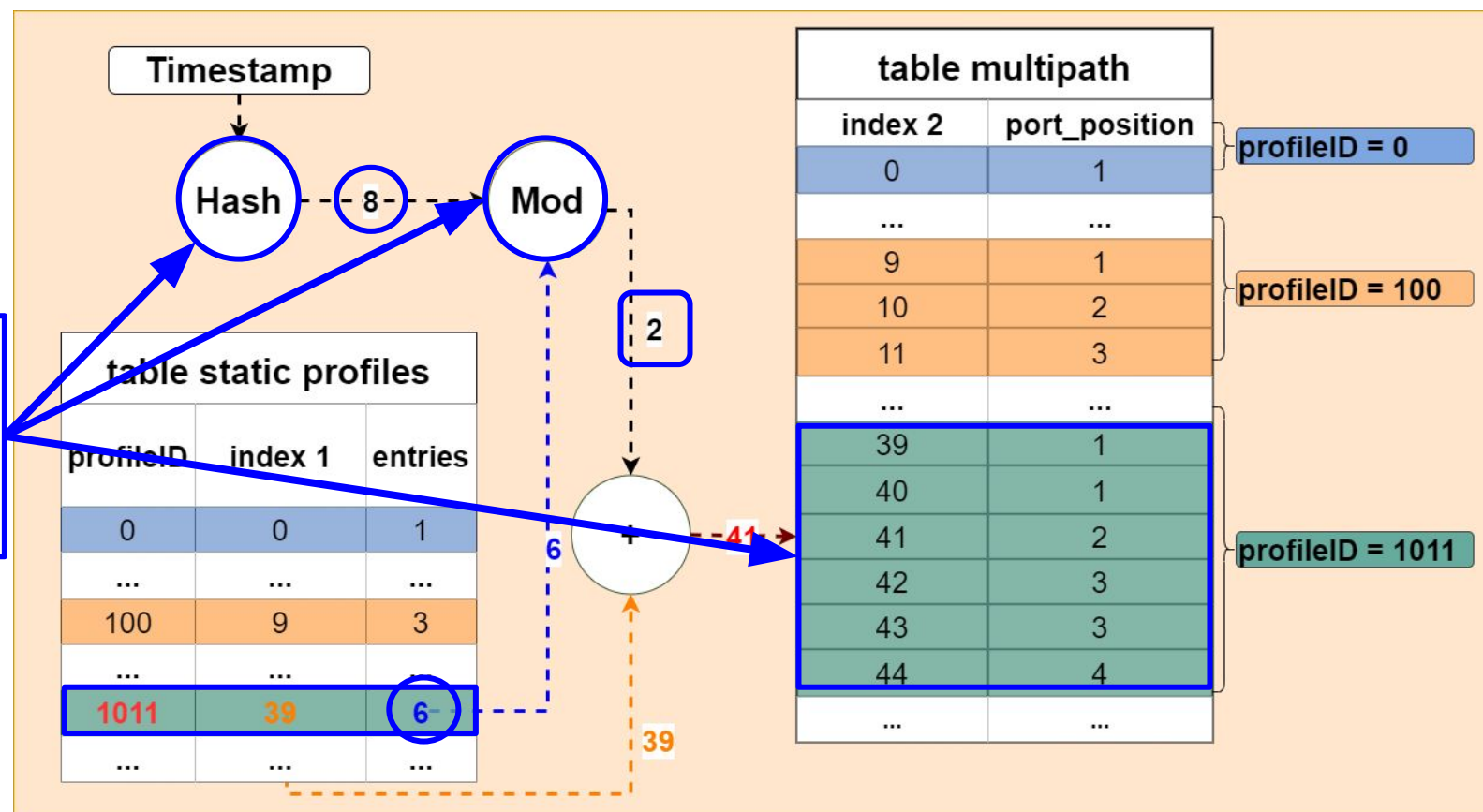


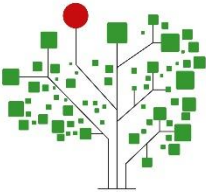


Como funciona o MTS-PolKA?

- Distribuição de Tráfego

1º é obtido o resto da divisão entre o hash do timestamp e quantidade de entrada para definir o intervalo

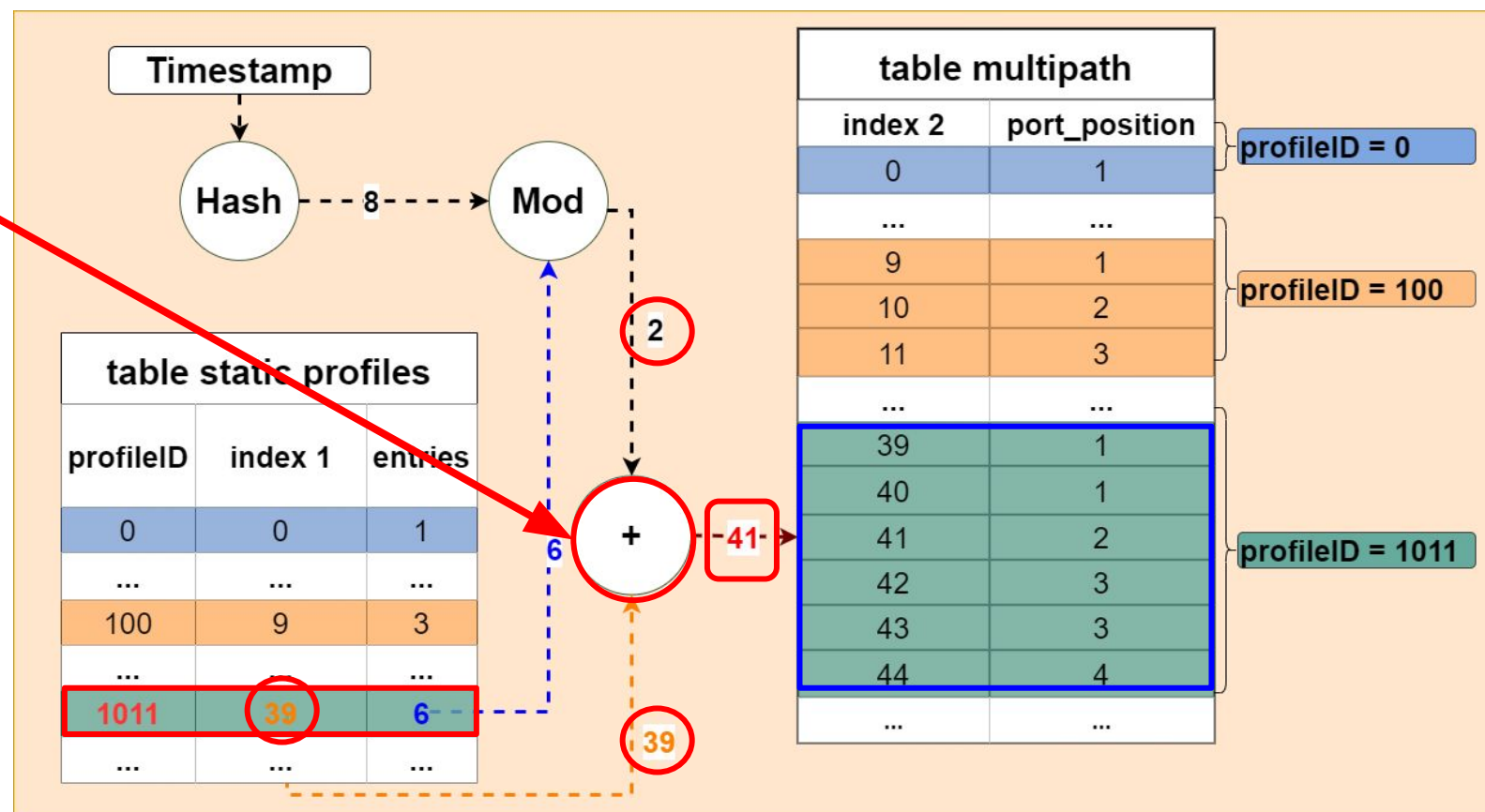


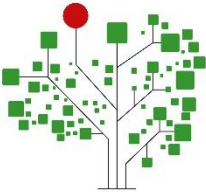


Como funciona o MTS-PolKA?

- Distribuição de Tráfego

2º Em seguida é realizada a soma entre o resultado do resto da divisão do hash com o index 1 para se obter a porta de saída.



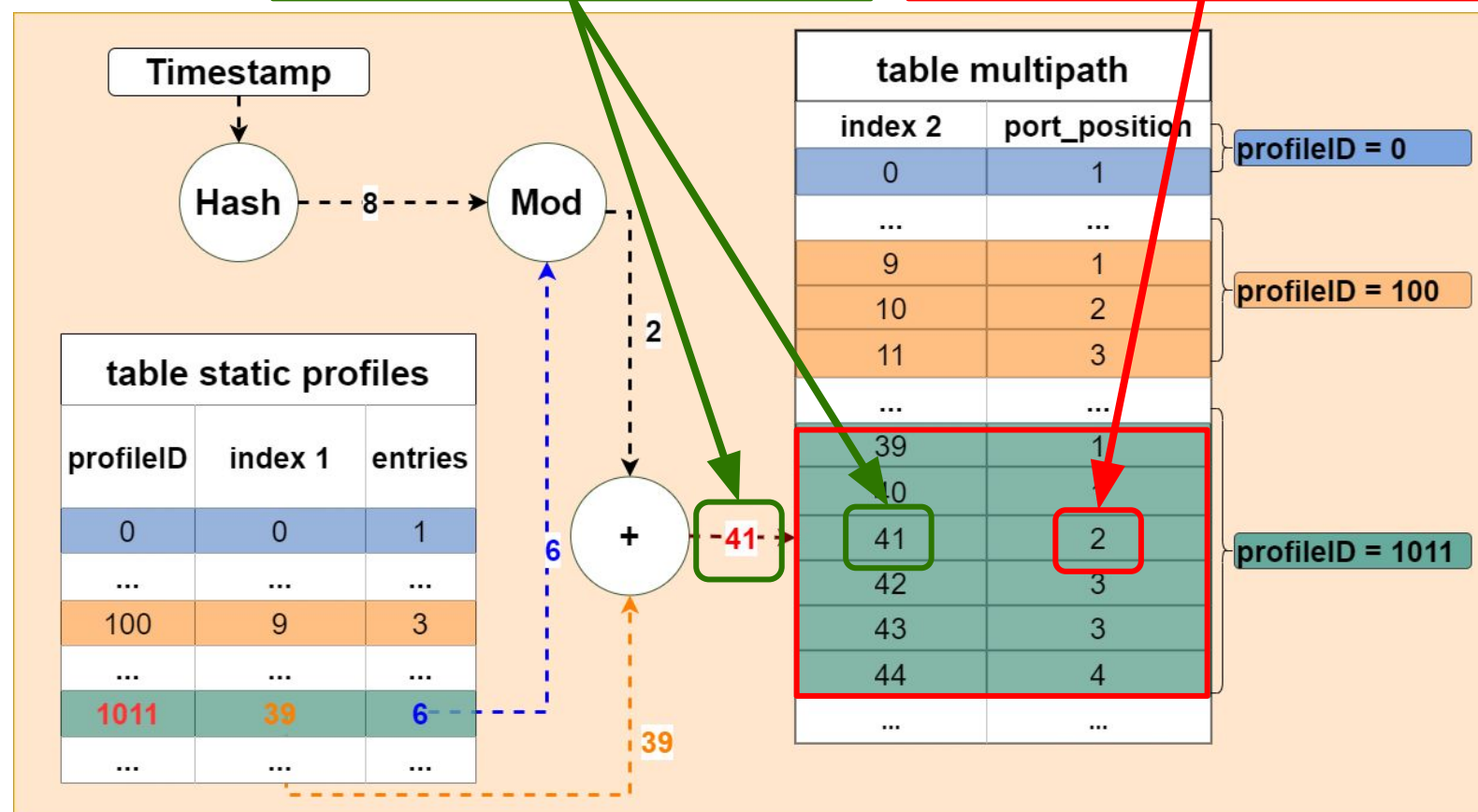


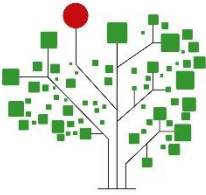
Como funciona o MTS-PolKA?

- Distribuição de Tráfego

Resultando no índice
index2 = 41 da table
multipath

O index 2 indica que a
segunda porta ativa
(*port position* = 2)





Como funciona o MTS-PolKA?

- Distribuição de Tráfego

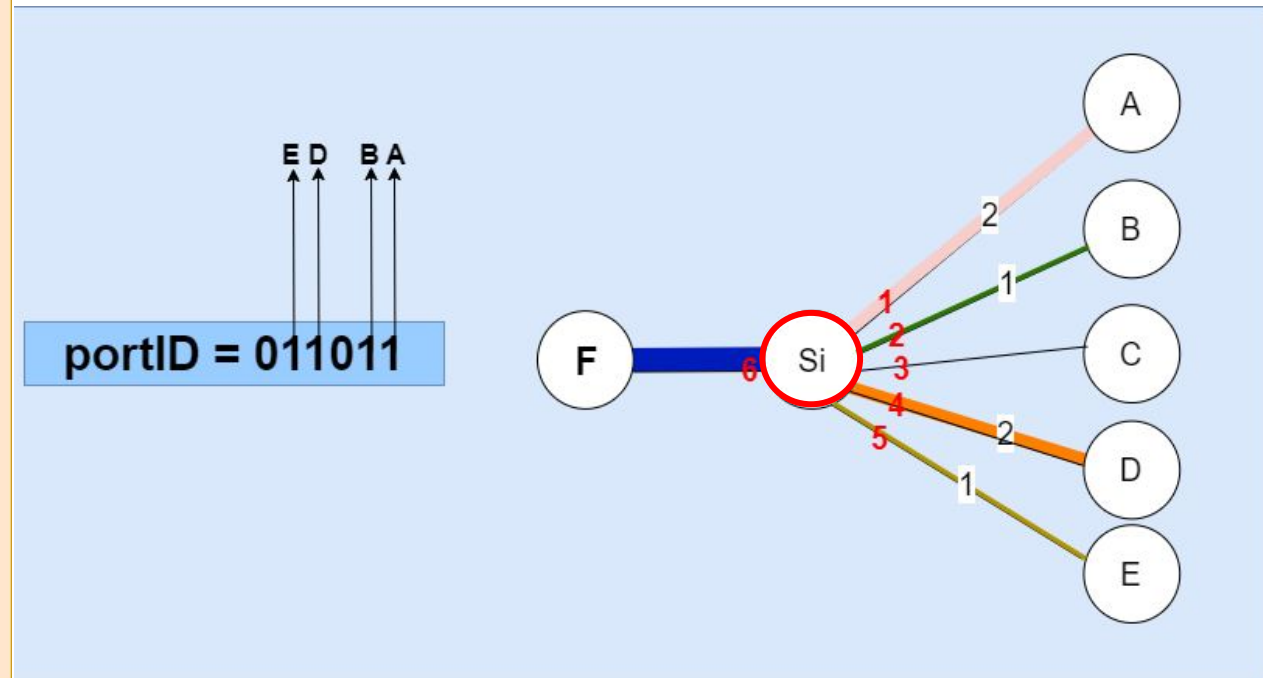
**Segunda
porta ativa
(port position
= 2)**

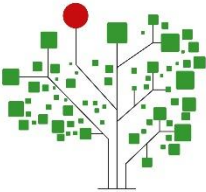
table multipath	
index 2	port_position
0	1
...	...
9	1
10	2
11	3
...	...
39	1
40	1
41	2
42	3
43	3
44	4
...	...

profileID = 0

profileID = 100

profileID = 1011





Como funciona o MTS-PolKA?

- Distribuição de Tráfego

**Segunda
porta ativa
(port position
= 2)**

table multipath	
index 2	port_position
0	1
...	...
9	1
10	2
11	3
...	...
39	1
40	1
41	2
42	3
43	3
44	4
...	...

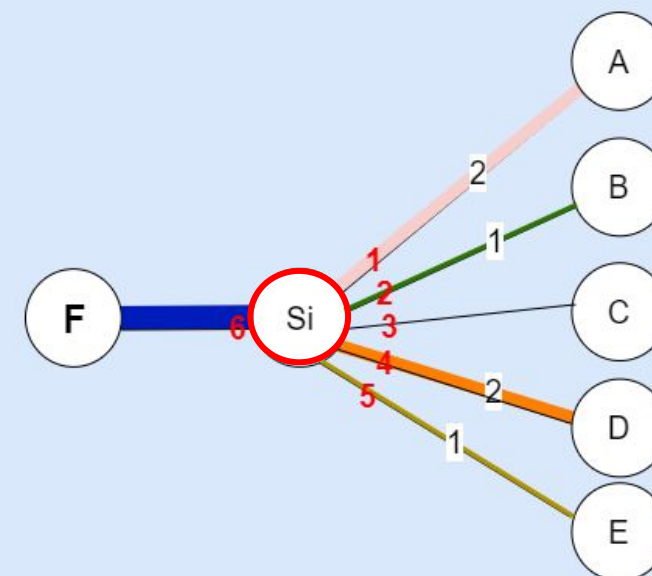
profileID = 0

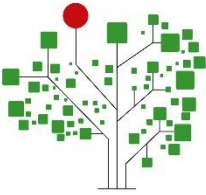
profileID = 100

profileID = 1011

portID = 011011 foi
previamente calculado na
etapa de definição da rota

portID = 011011





Como funciona o MTS-PoIKA?

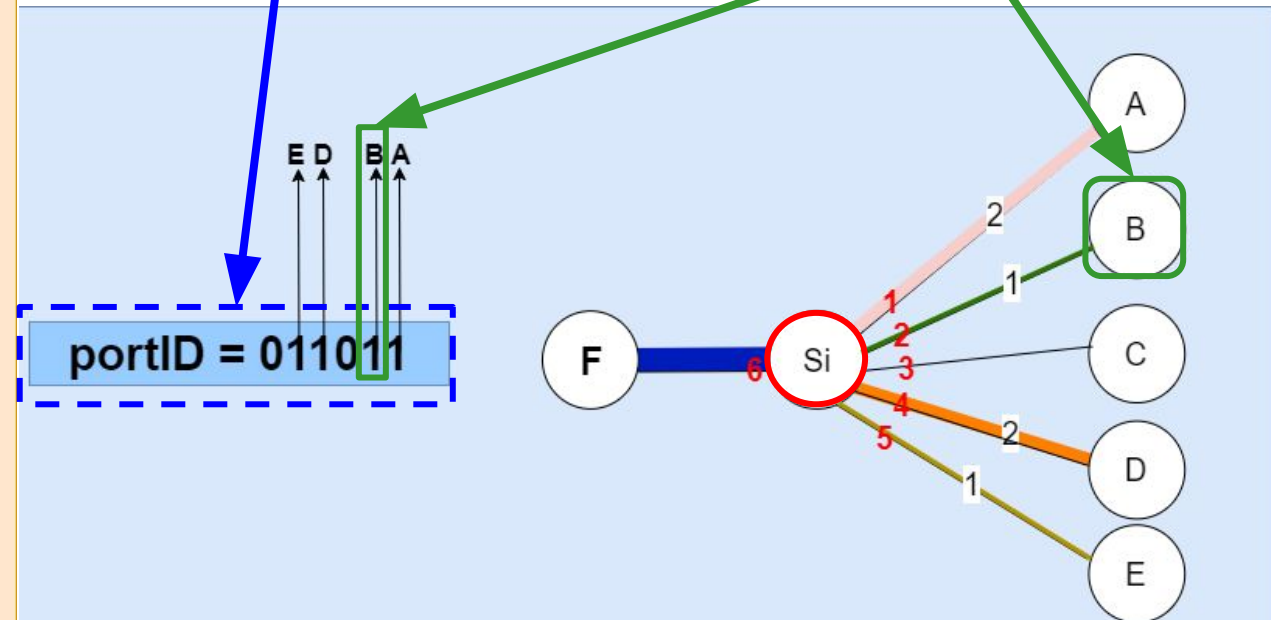
- Distribuição de Tráfego

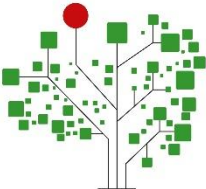
portID = 011011 foi
previamente calculado
na etapa de definição
da rota

Porta 2 (segunda porta
ativa) representa o
caminho de saída do
pacote para o nó B.

Segunda
porta ativa
(port position
= 2)

table multipath	
index 2	port_position
0	1
...	...
9	1
10	2
11	3
...	...
39	1
40	1
41	2
42	3
43	3
44	4
...	...

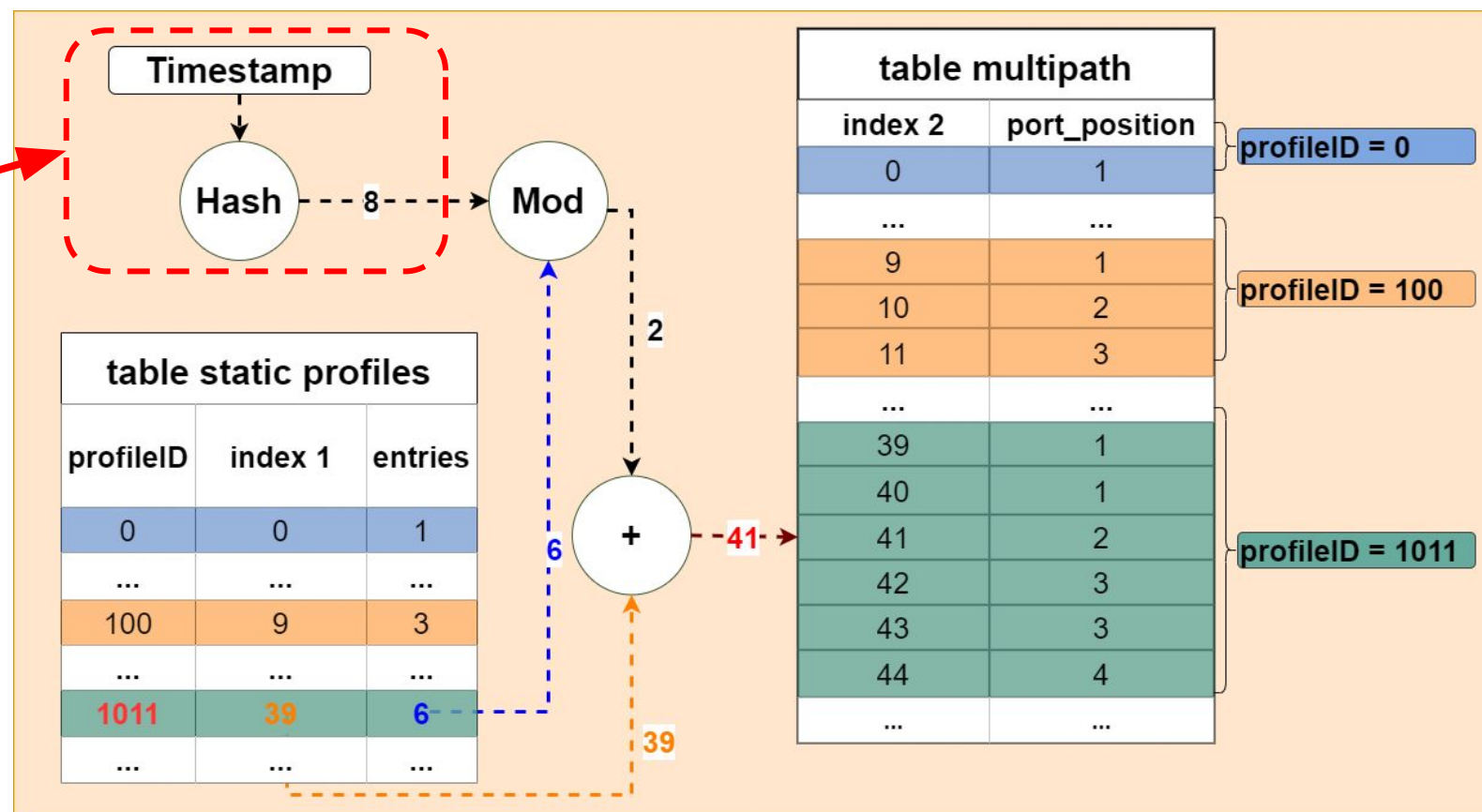


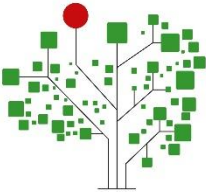


Como funciona o MTS-PolKA?

- Aplicando o algoritmo de seleção de portas

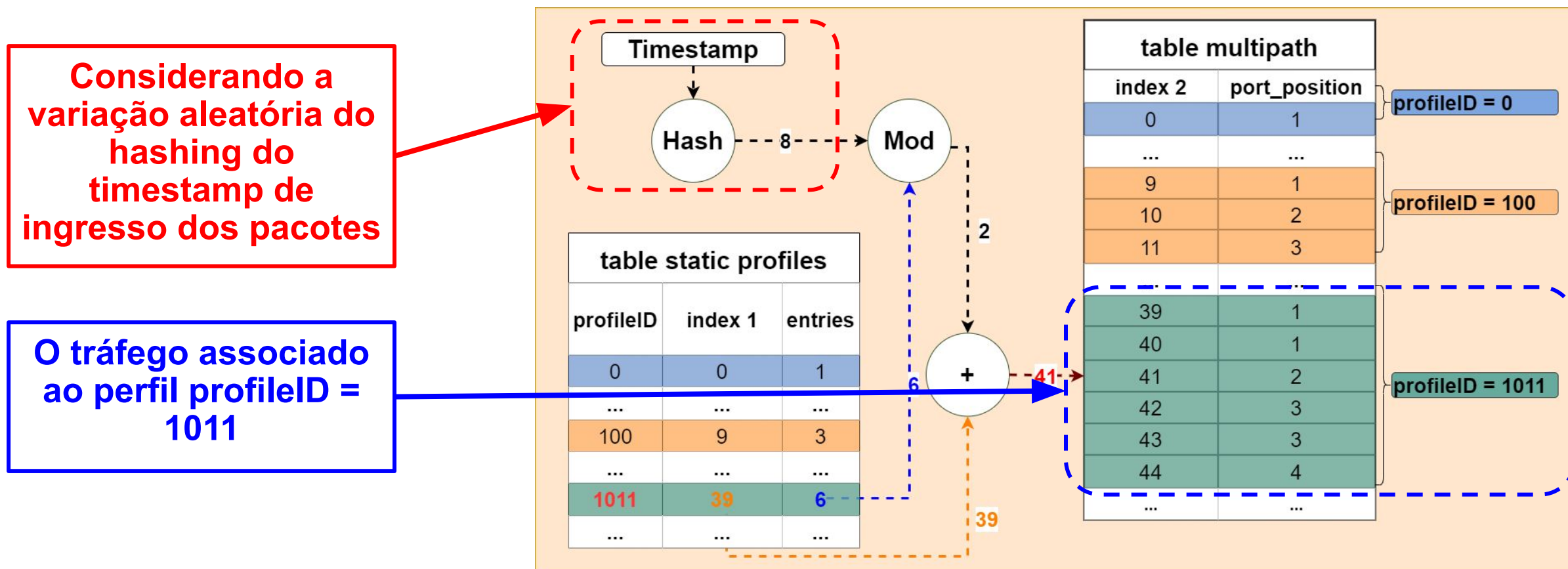
Considerando a
variação aleatória do
hashing do
timestamp de
ingresso dos pacotes

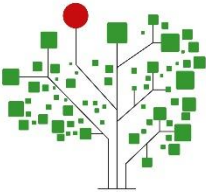




Como funciona o MTS-PolKA?

- Aplicando o algoritmo de seleção de portas





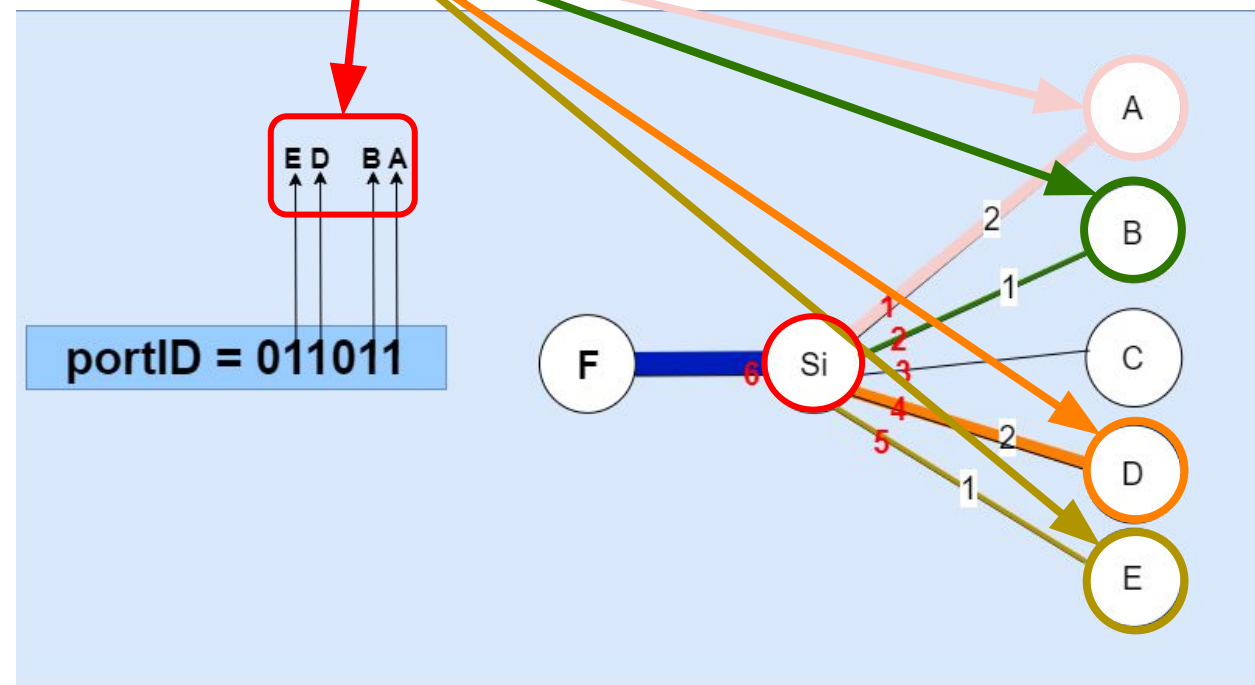
Como funciona o MTS-PolKA?

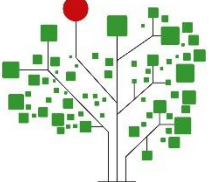
- Aplicando o algoritmo de seleção de portas

O tráfego
associado ao
perfil profileID
= 1011

table multipath	
index 2	port_position
0	1
...	...
9	1
10	2
11	3
...	...
39	1
40	1
41	2
42	3
43	3
44	4
...	...

Portas ativas = ABDE





PPComp

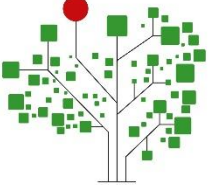
Mestrado Profissional em
Computação Aplicada



INSTITUTO FEDERAL
Espírito Santo
Campus Serra

Avaliação Experimental

- Ambiente de desenvolvimento e testes:
 - Uso do emulador Mininet para conduzir experimentos.
 - Capacidade de compilação de programas P4-16 e configuração de switches bmv2.



PPComp

Mestrado Profissional em
Computação Aplicada

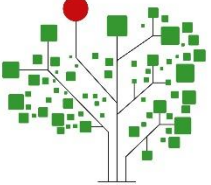


INSTITUTO FEDERAL

Espírito Santo
Campus Serra

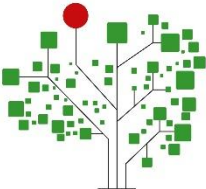
Avaliação Experimental

- Foram realizados três experimentos:
- Na apresentação colocamos alguns testes, o restante ficou no artigo:
 - Um teste funcional, para demonstrar a aplicação de diferentes perfis de divisão de tráfego;
 - Um teste com múltiplos fluxos concorrentes.



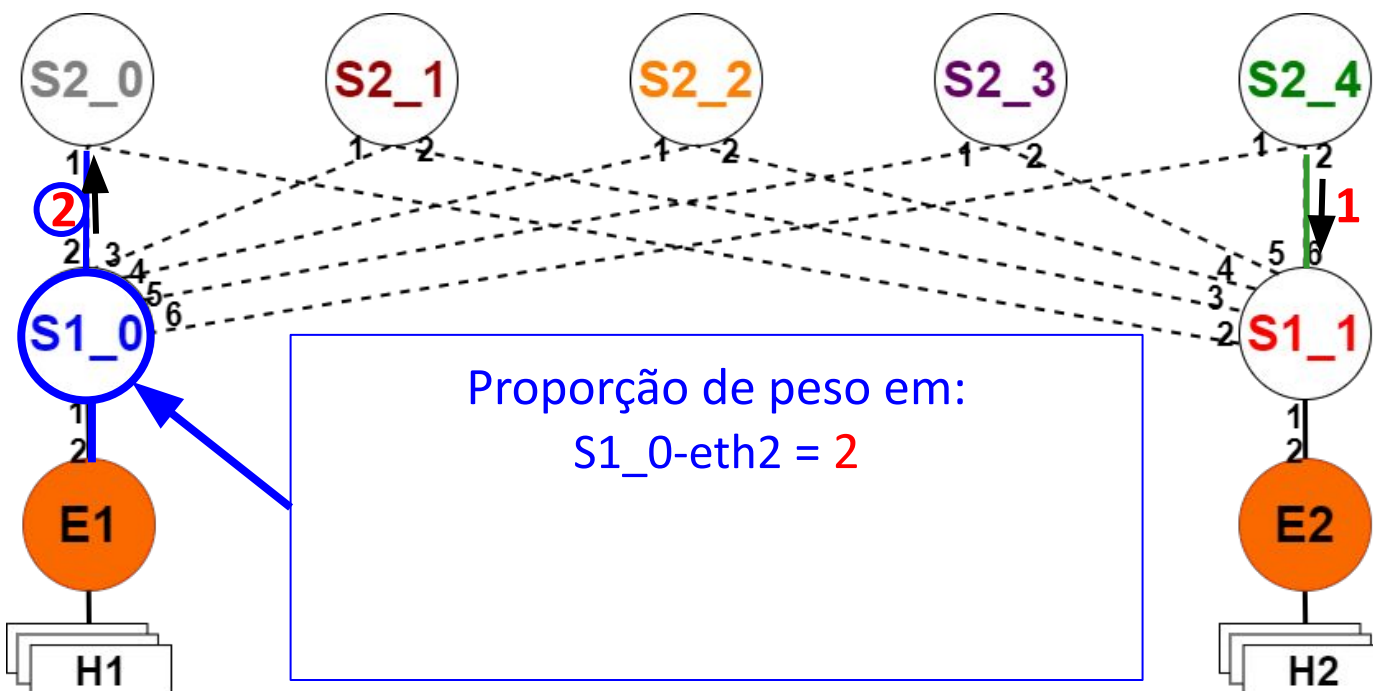
Avaliação Experimental

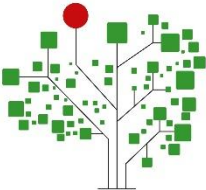
- Teste 1: Validação Funcional
 - Três diferentes perfis de tráfego foram selecionados na origem para um switch de núcleo específico (S1_0):
 - Iremos explicar apenas o Teste 1: perfil 1011 (binário) e 11 (decimal).



Avaliação Experimental

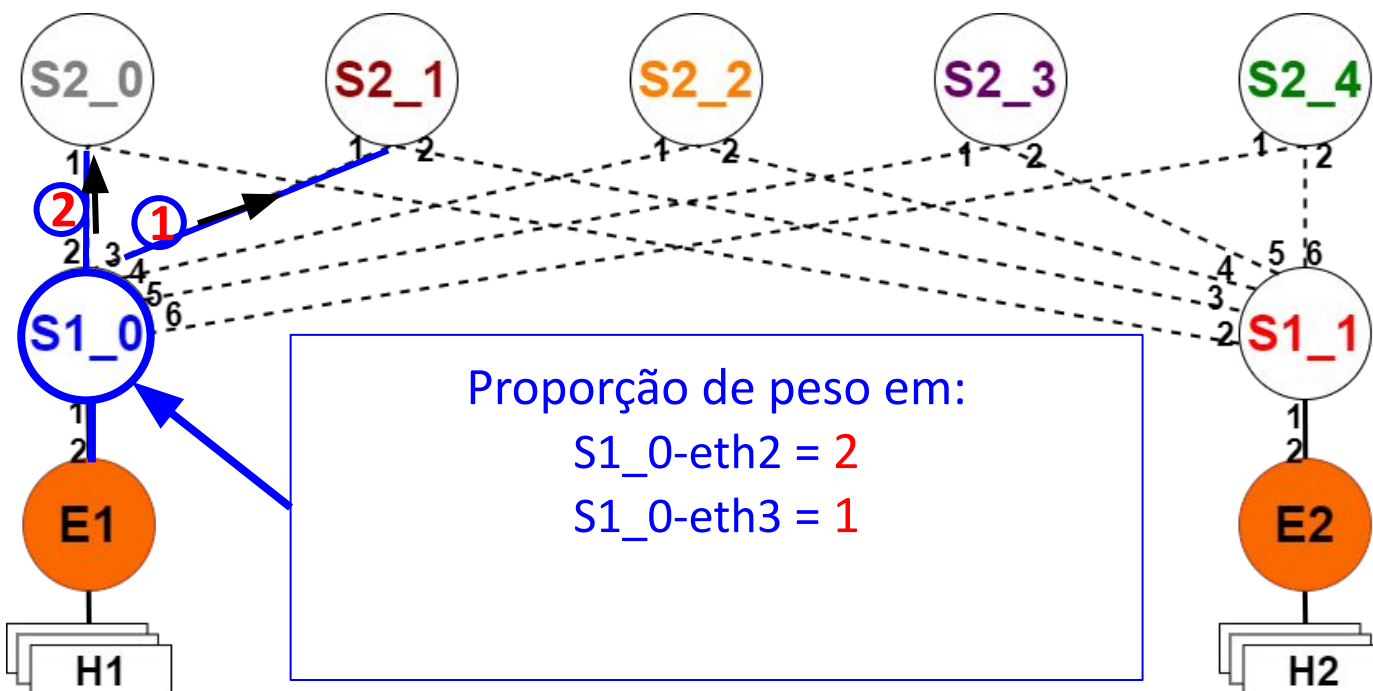
- Teste 1: Validação Funcional - Perfil 1011

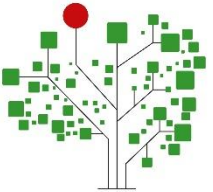




Avaliação Experimental

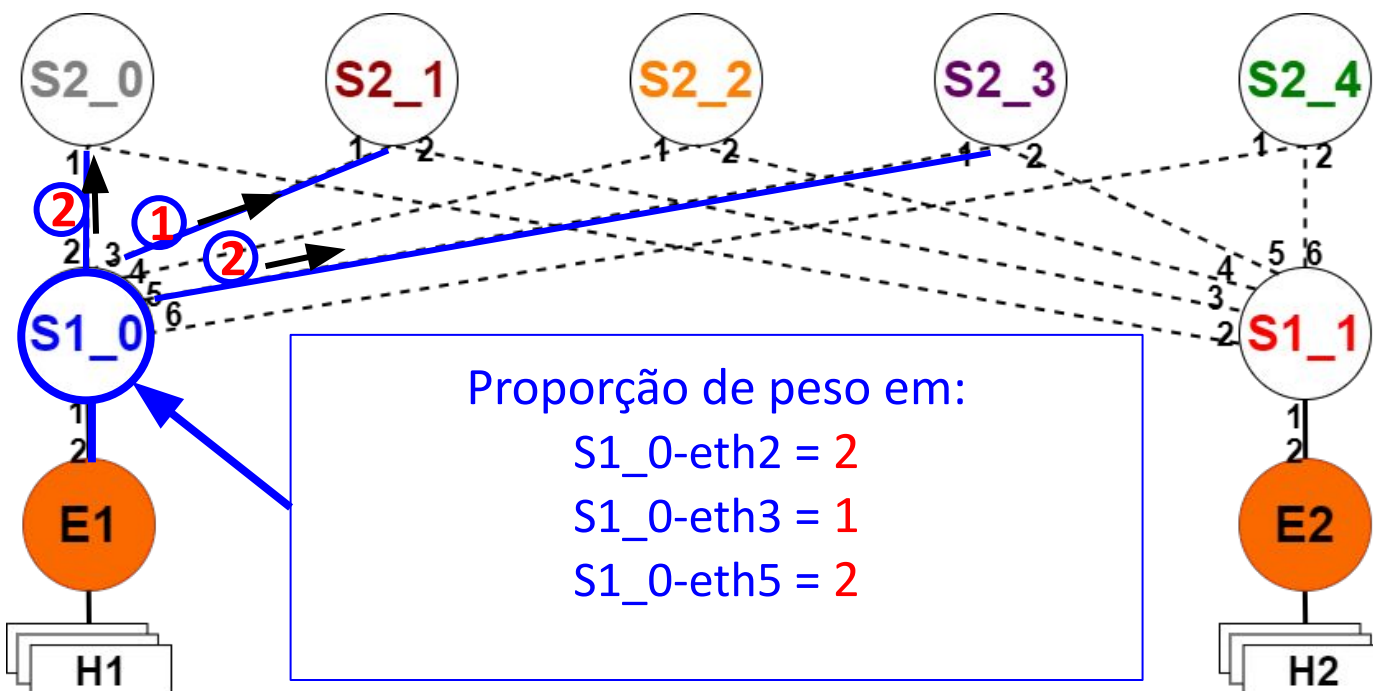
- Teste 1: Validação Funcional - Perfil 1011

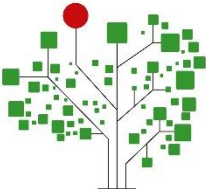




Avaliação Experimental

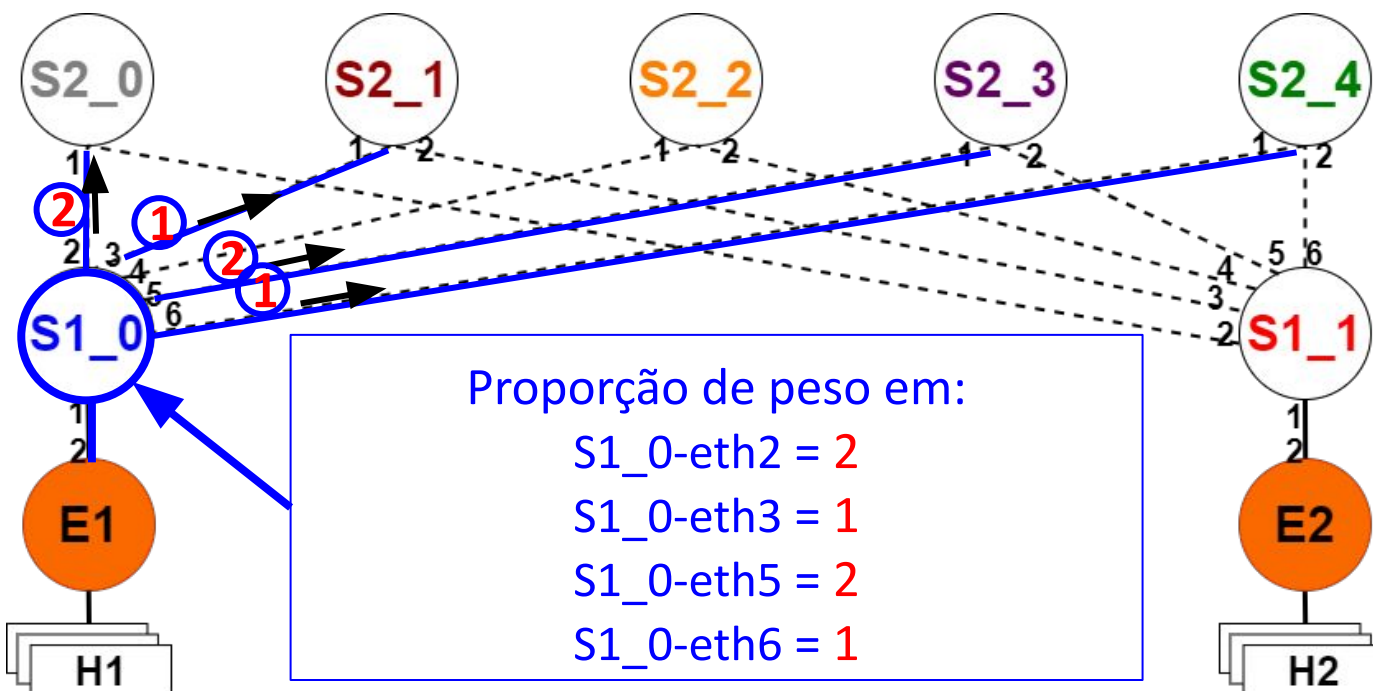
- Teste 1: Validação Funcional - Perfil 1011

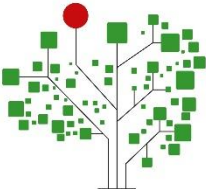




Avaliação Experimental

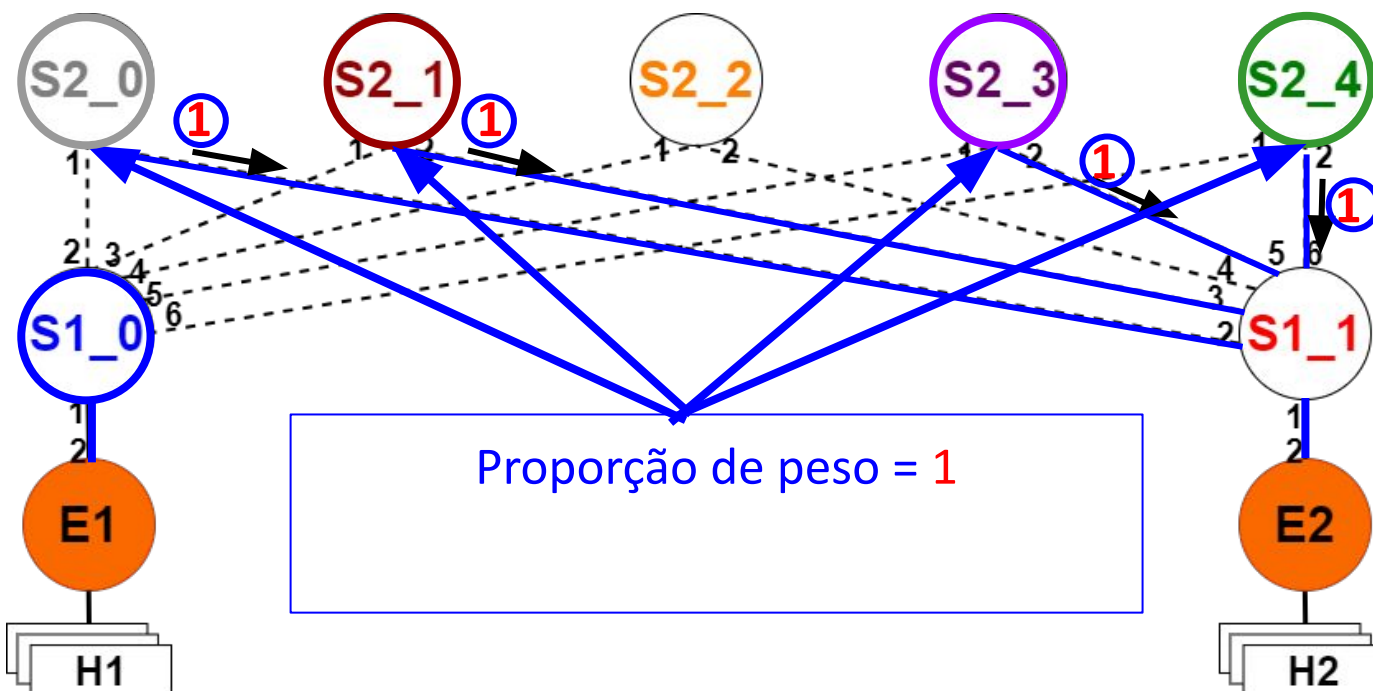
- Teste 1: Validação Funcional - Perfil 1011

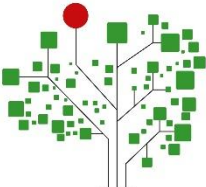




Avaliação Experimental

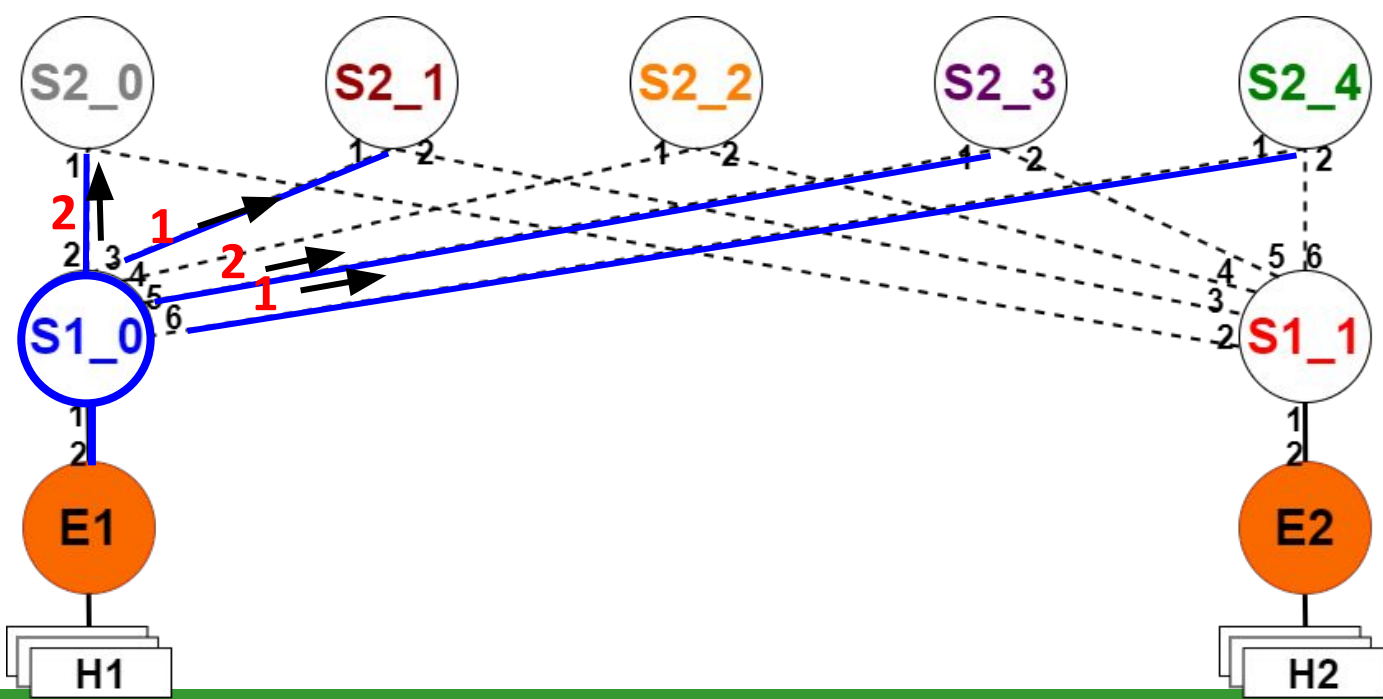
- Teste 1: Validação Funcional - Perfil 1011



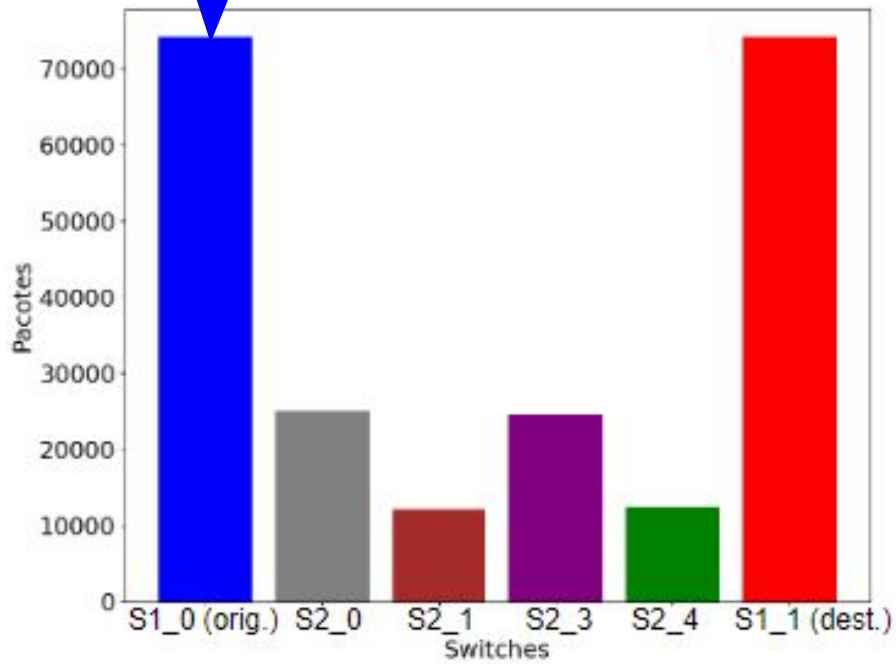


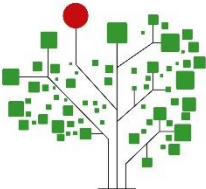
Avaliação Experimental

- Teste 1: Validação Funcional - Perfil 1011



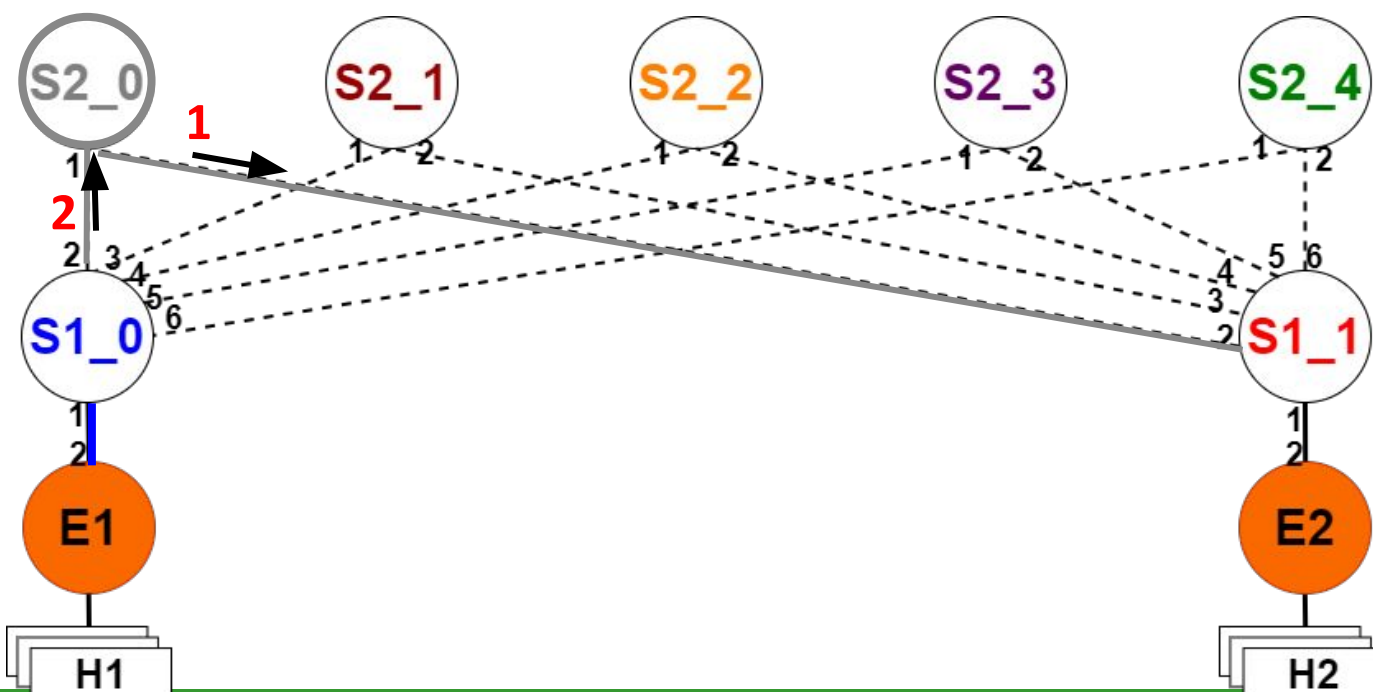
Todo o
tráfego de
S1_0 (Orig)



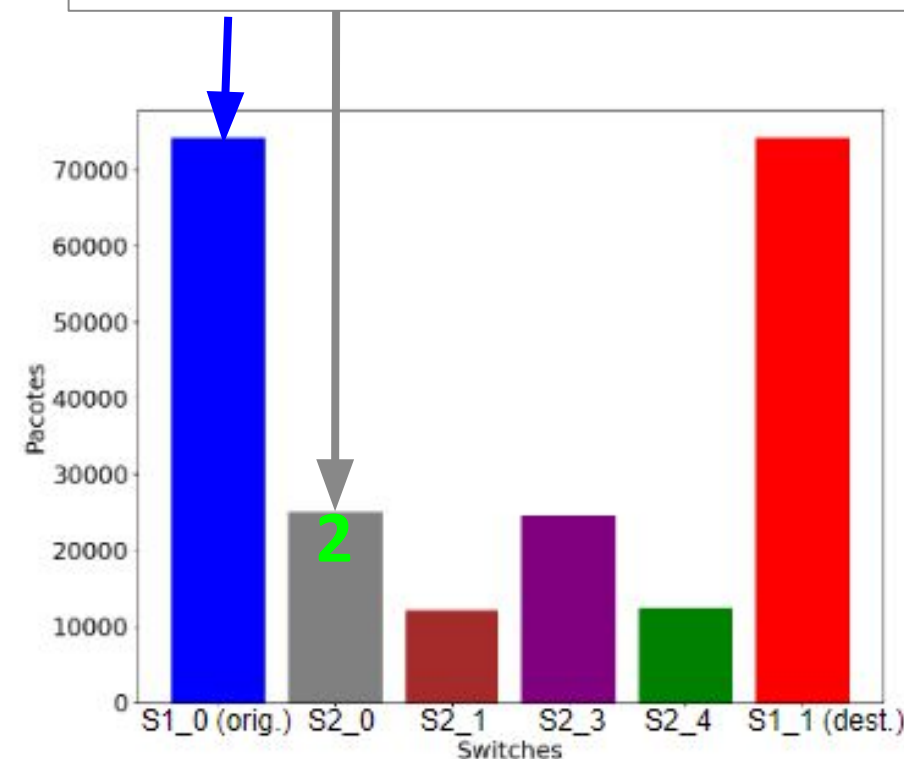


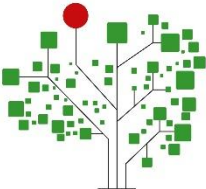
Avaliação Experimental

- Teste 1: Validação Funcional - Perfil 1011



Distribuiu o tráfego de forma variada em suas quatro portas ativas S2_0.

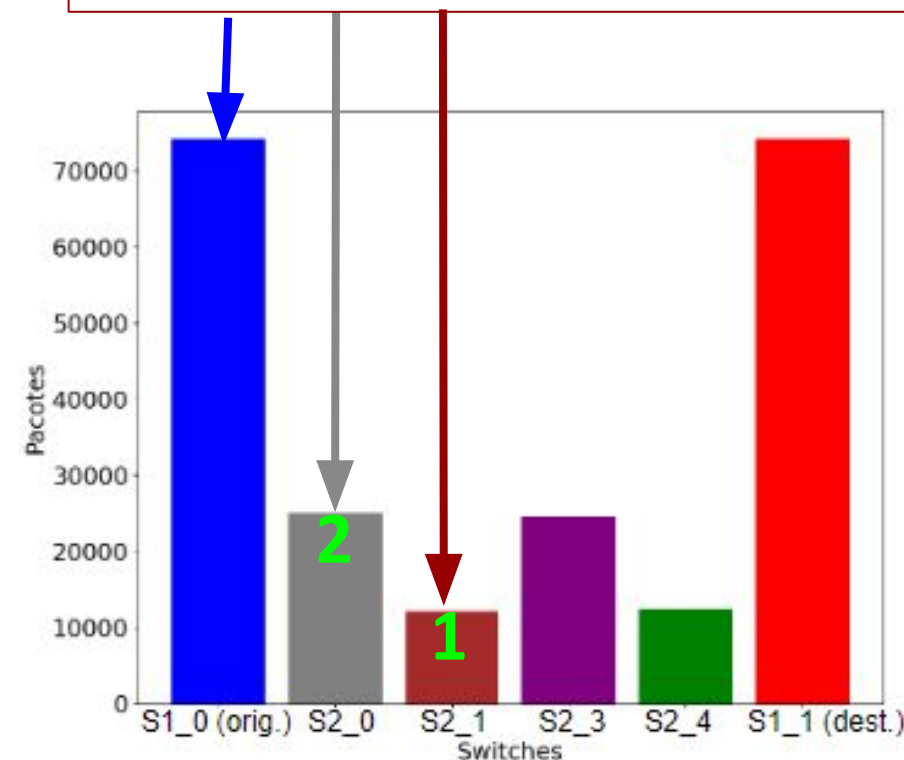
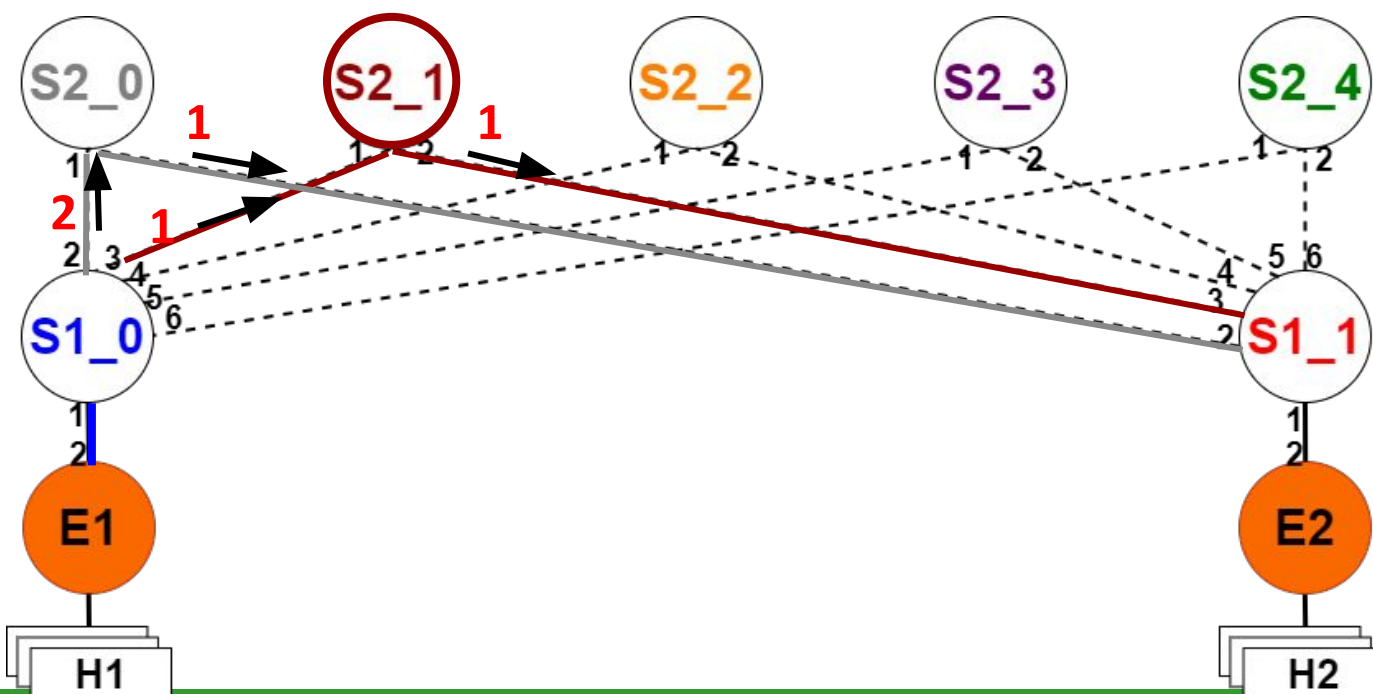


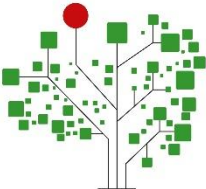


Avaliação Experimental

- Teste 1: Validação Funcional - Perfil 1011

Distribuiu o tráfego de forma variada em suas quatro portas ativas S2_1.

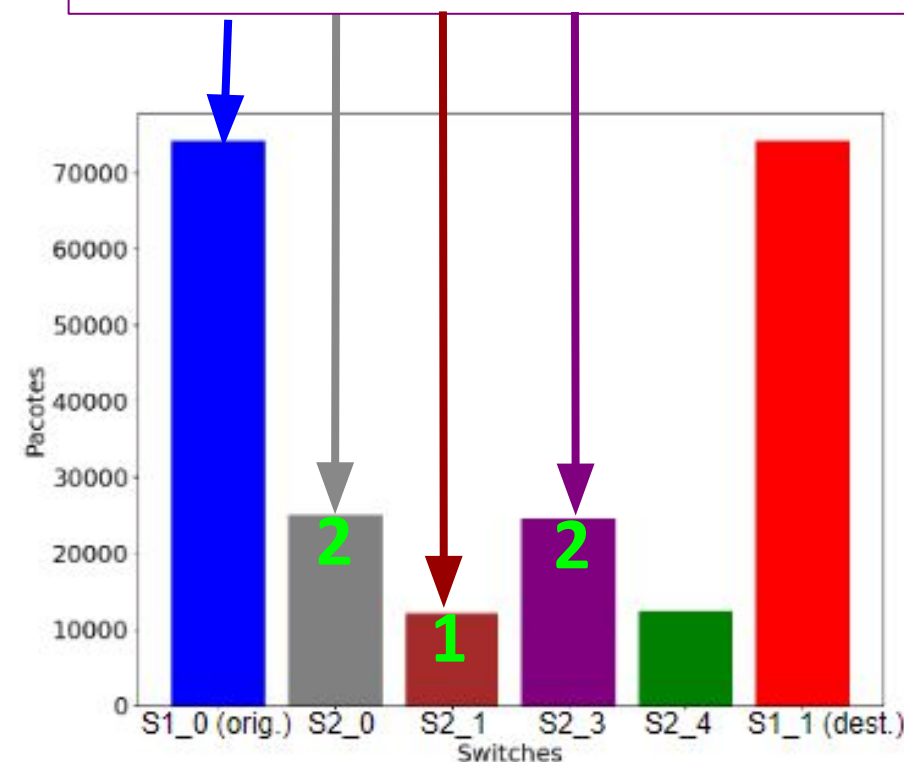
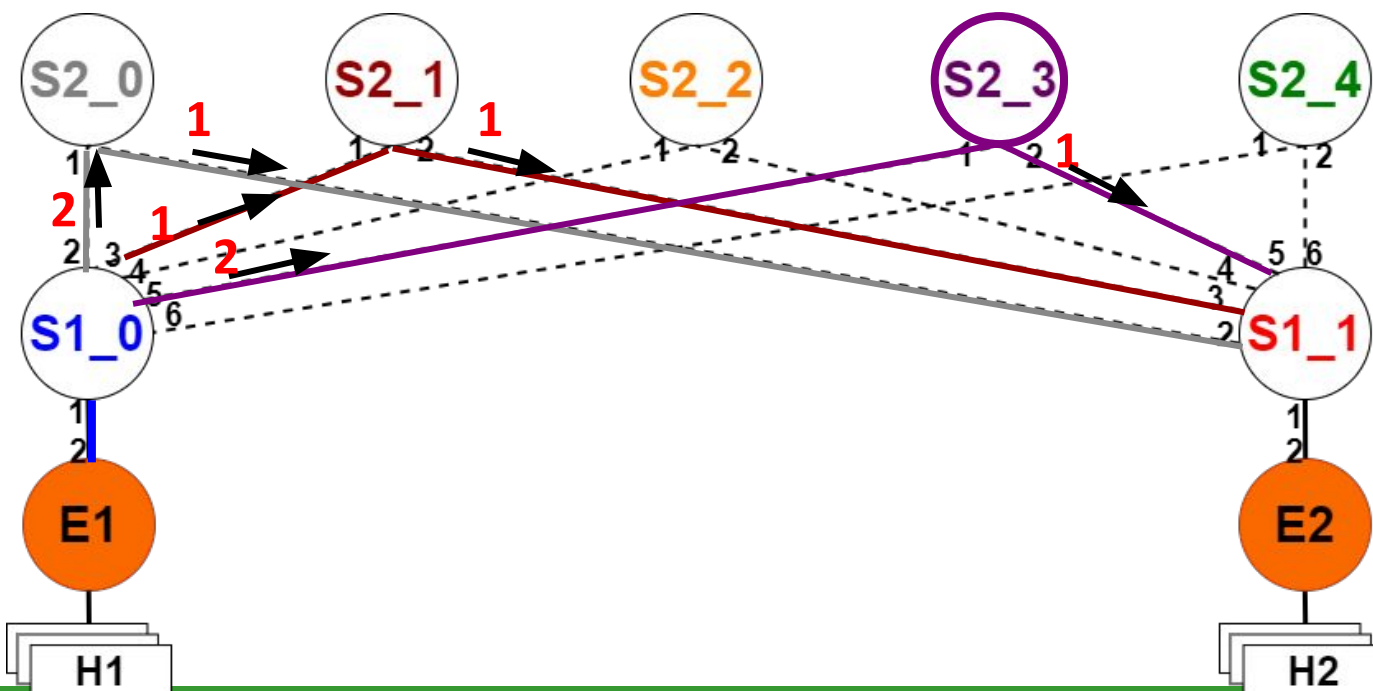


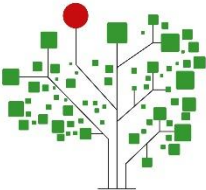


Avaliação Experimental

- Teste 1: Validação Funcional - Perfil 1011

Distribuiu o tráfego de forma variada em suas quatro portas ativas S2_3.

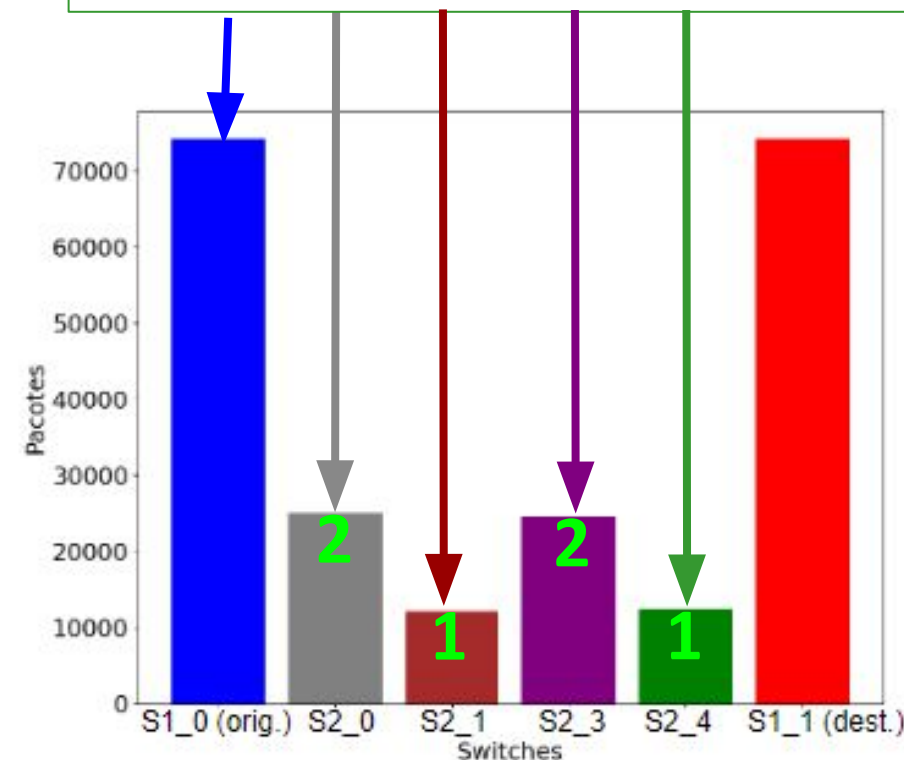
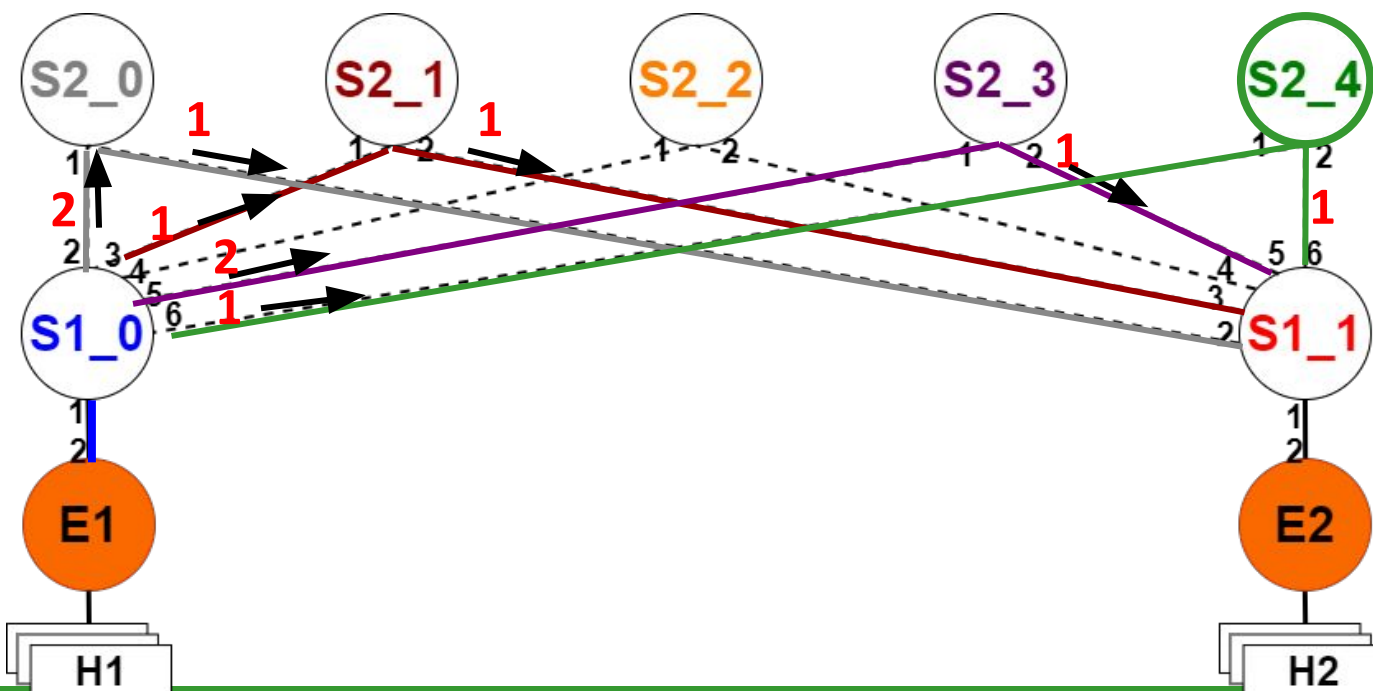


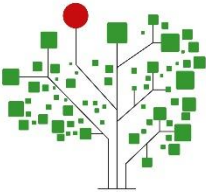


Avaliação Experimental

- Teste 1: Validação Funcional - Perfil 1011

Distribuiu o tráfego de forma variada em suas quatro portas ativas S2_4.

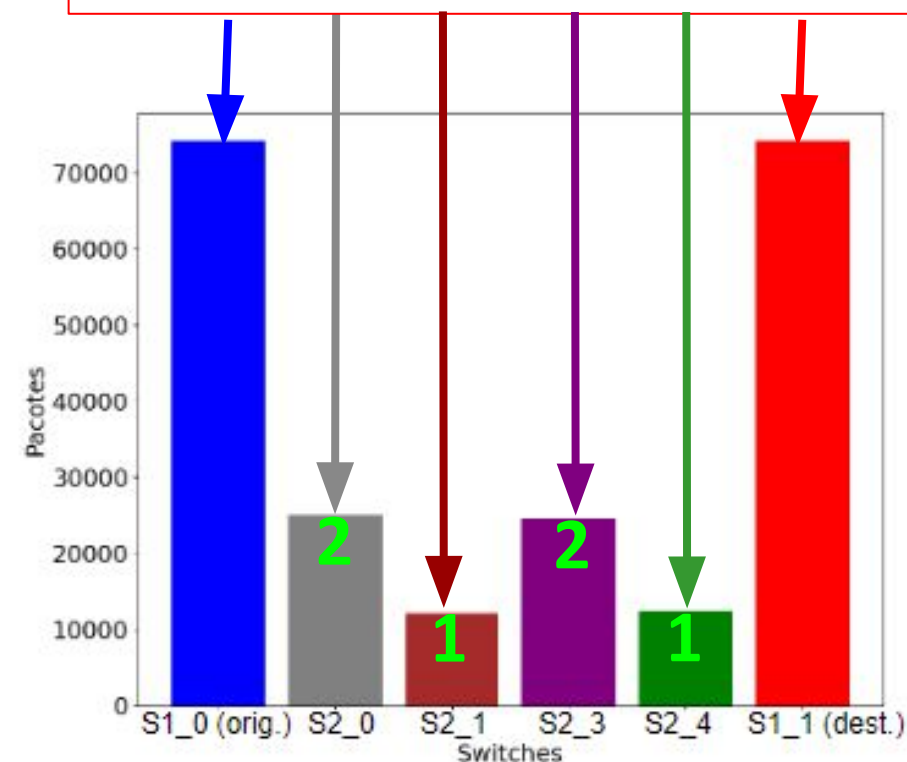
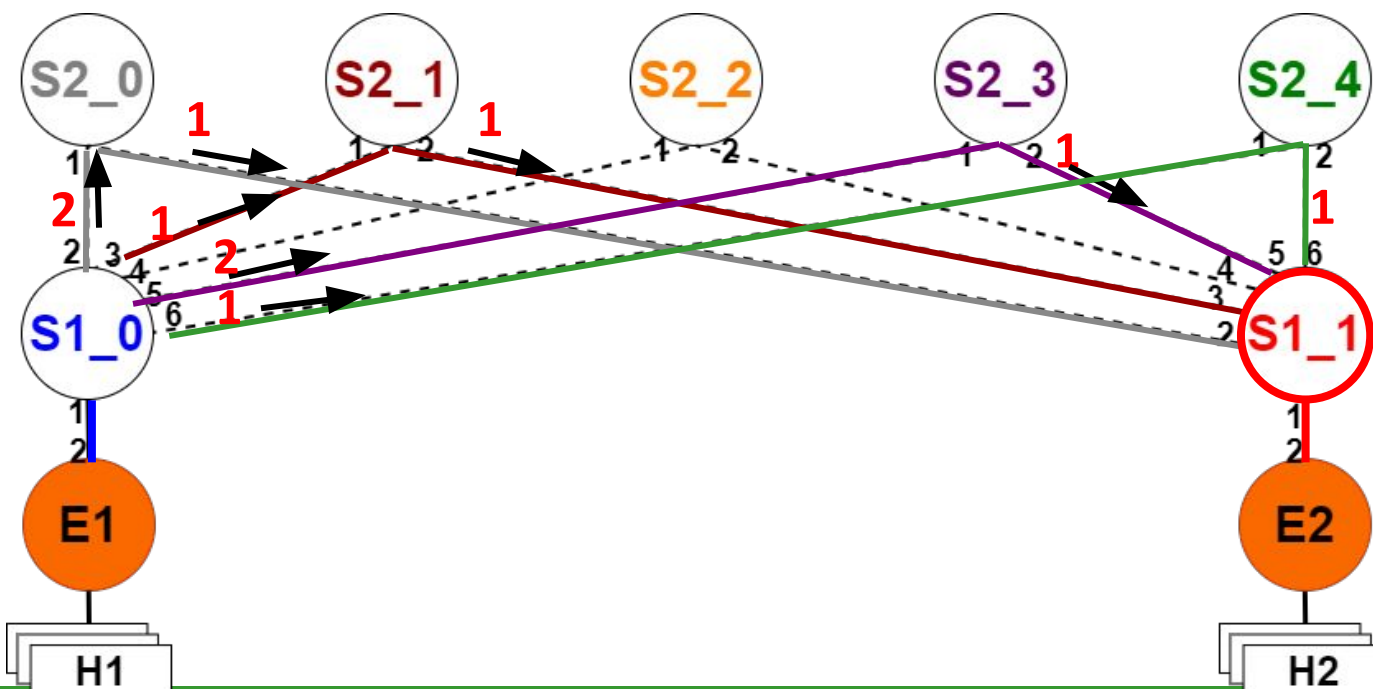


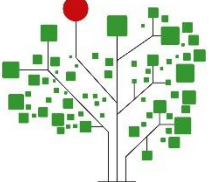


Avaliação Experimental

- Teste 1: Validação Funcional - Perfil 1011

Distribuiu o tráfego de forma variada em suas quatro portas ativas S1_1 (dest).





PPComp

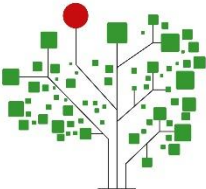
Mestrado Profissional em
Computação Aplicada



INSTITUTO FEDERAL
Espírito Santo
Campus Serra

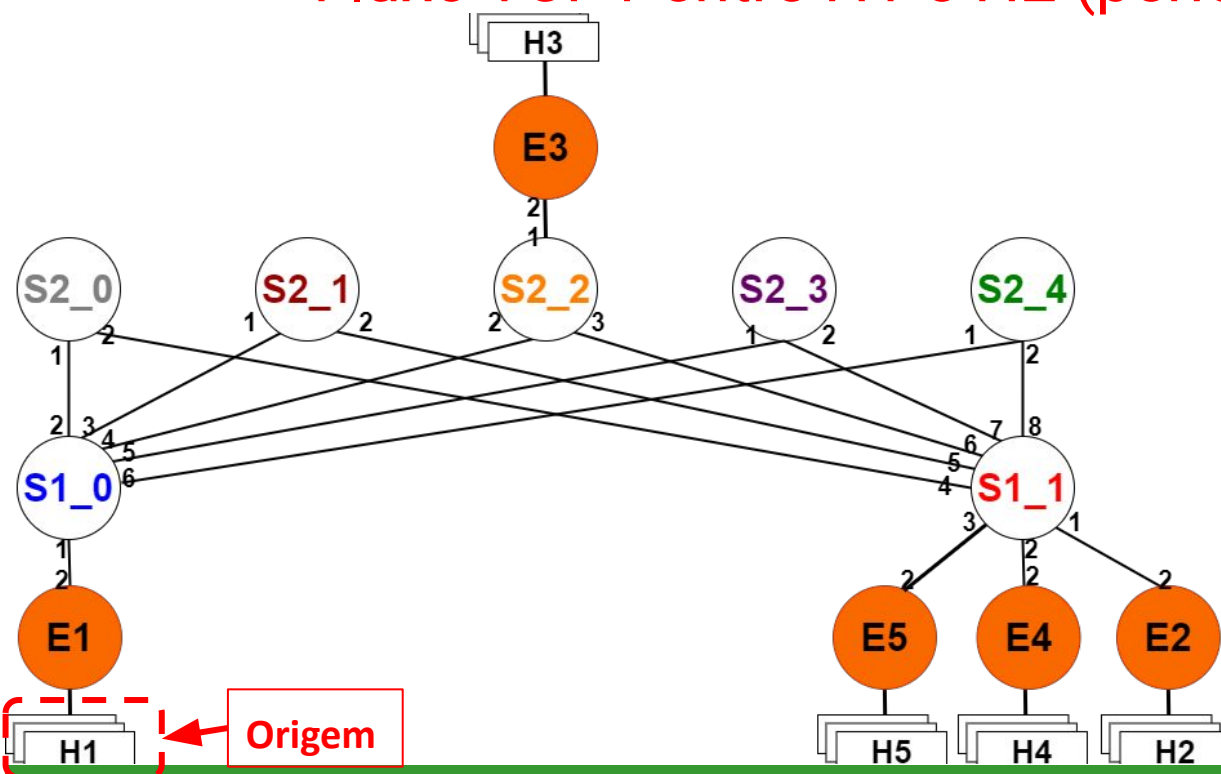
Avaliação Experimental

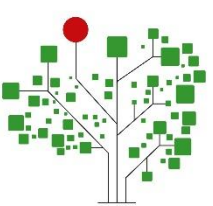
- Teste 3: Múltiplos fluxos e concorrência
 - Este experimento demonstra o funcionamento de múltiplos fluxos e a migração ágil de perfil de tráfego para otimizar o uso de recursos com concorrência.



Avaliação Experimental

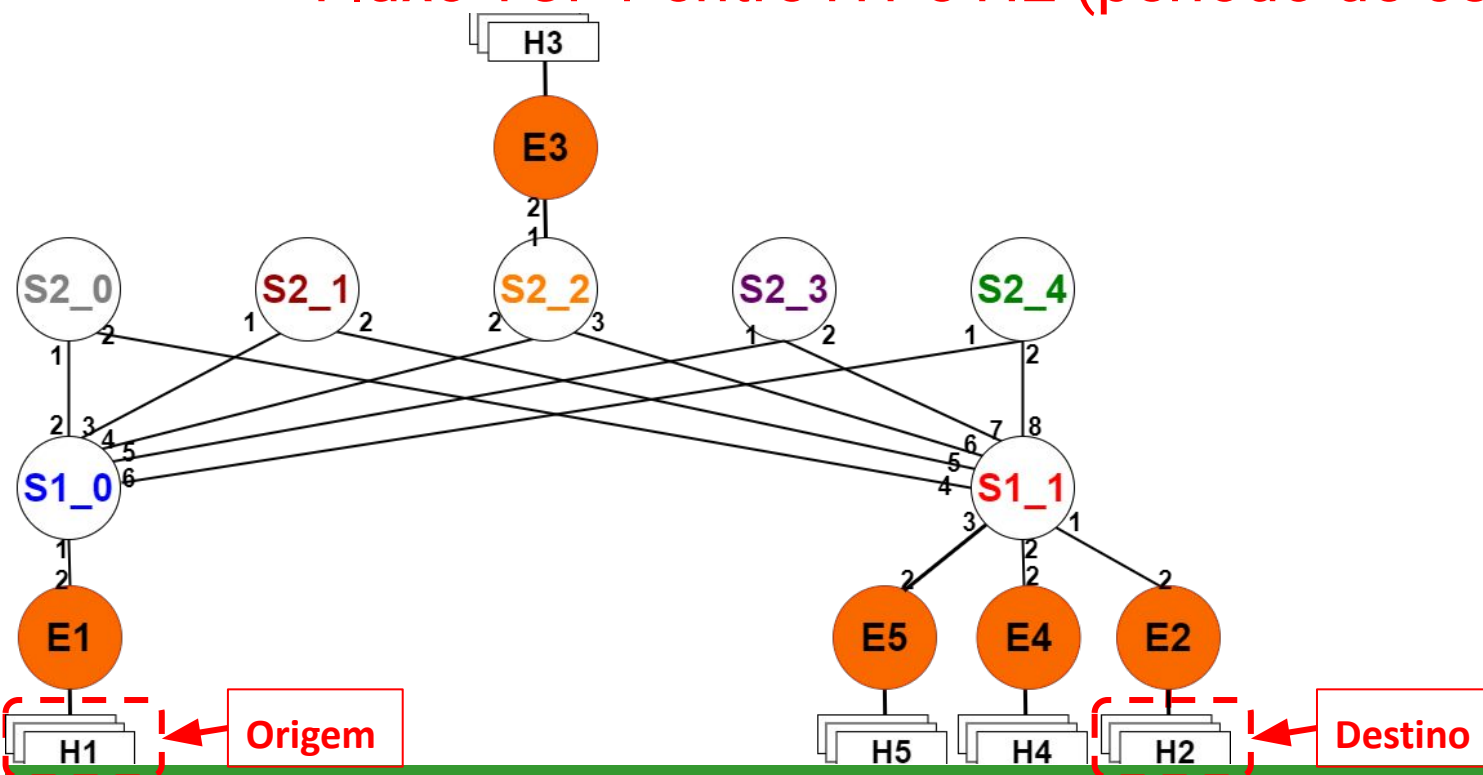
- Teste 3: Múltiplos fluxos e concorrência
 - Fluxo TCP1 entre H1 e H2 (período de 0s a 10s);

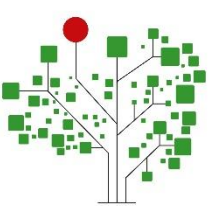




Avaliação Experimental

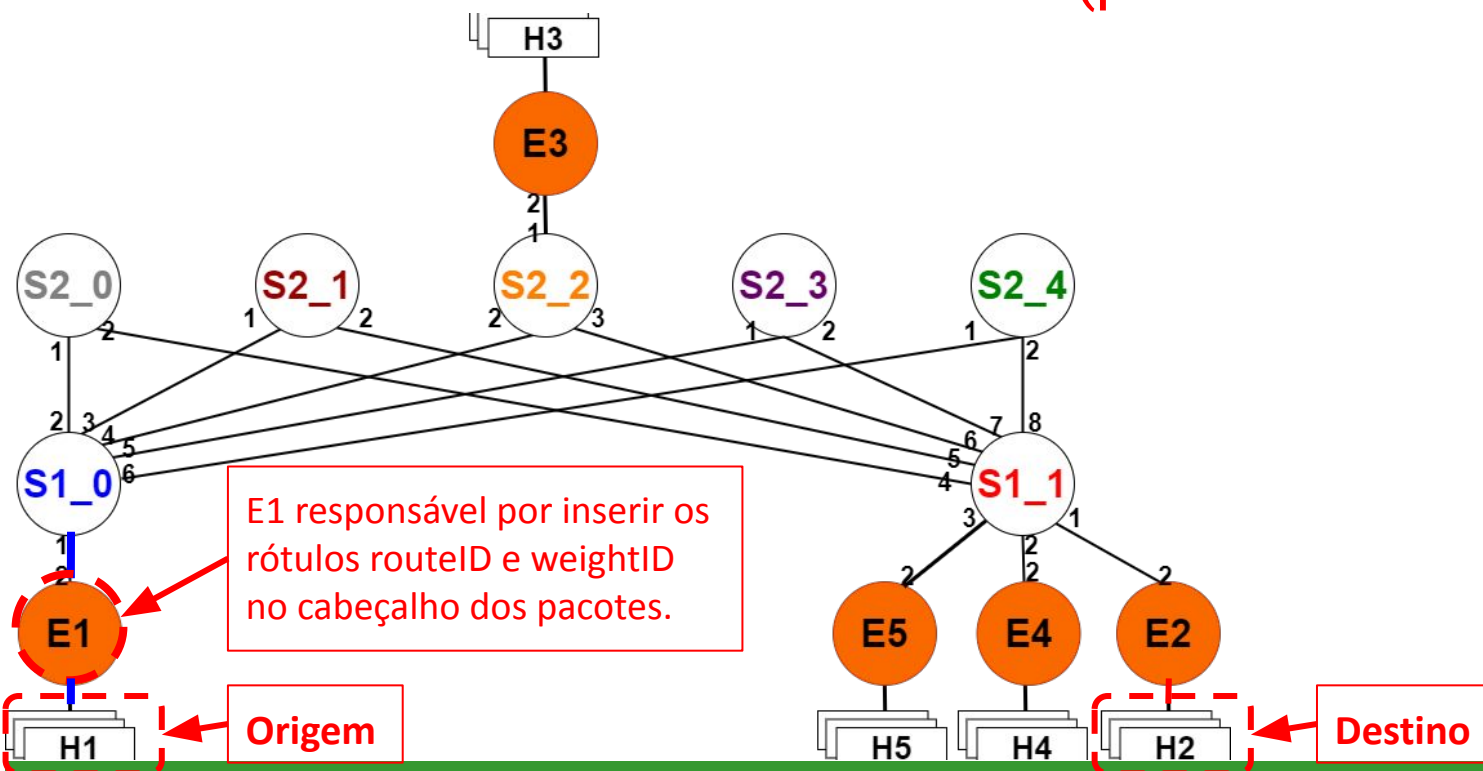
- Teste 3: Múltiplos fluxos e concorrência
 - Fluxo TCP1 entre H1 e H2 (período de 0s a 10s);

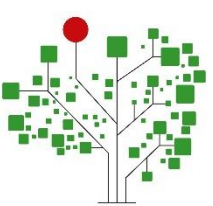




Avaliação Experimental

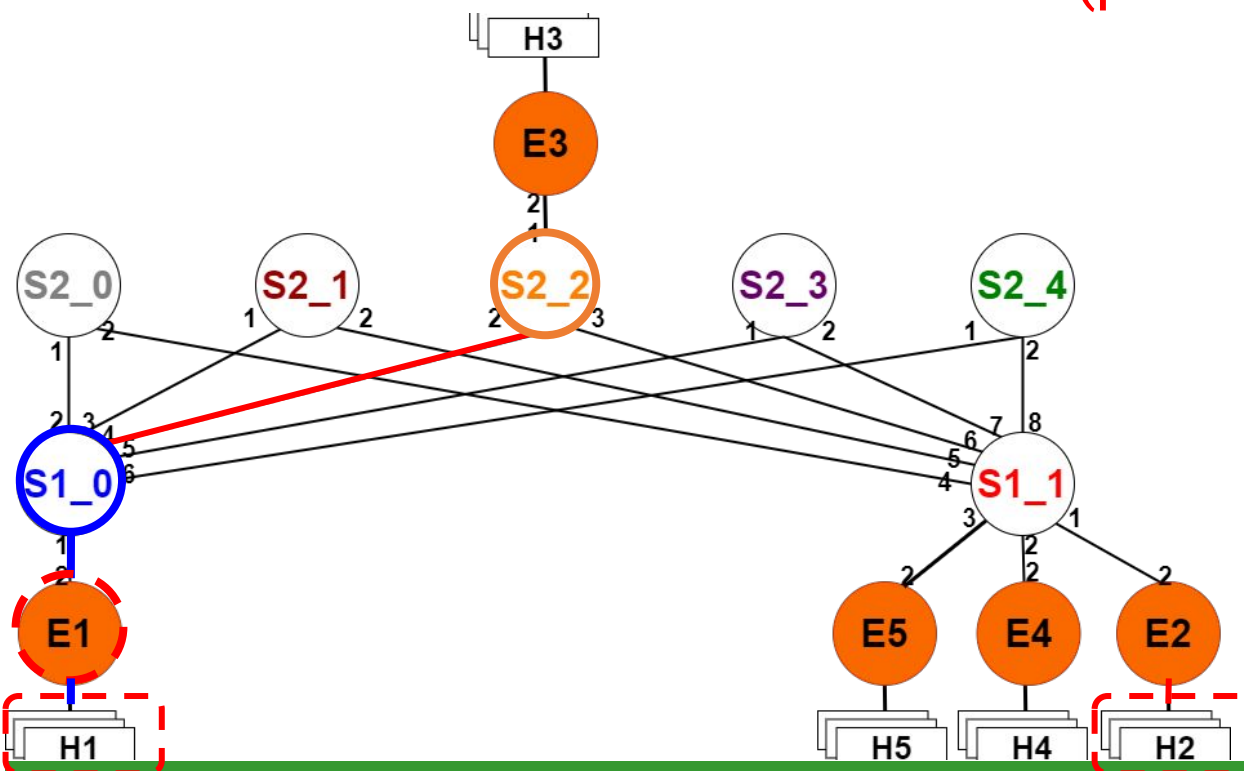
- Teste 3: Múltiplos fluxos e concorrência
 - Fluxo TCP1 entre H1 e H2 (período de 0s a 10s);

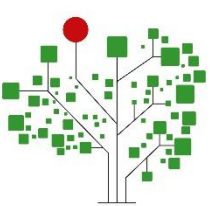




Avaliação Experimental

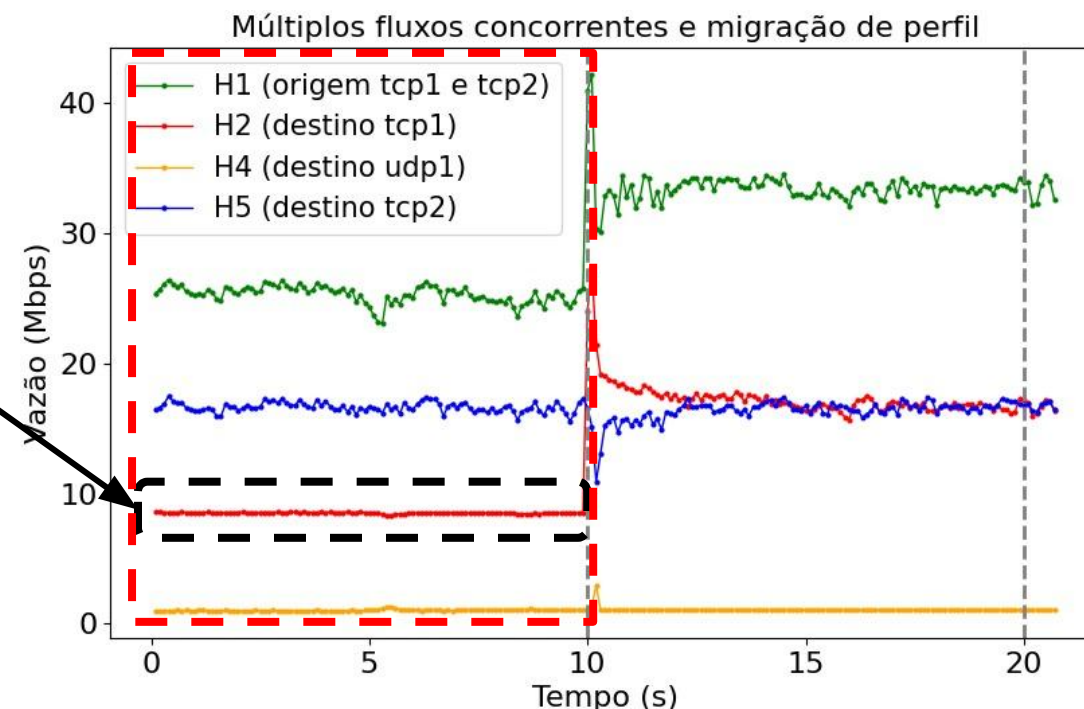
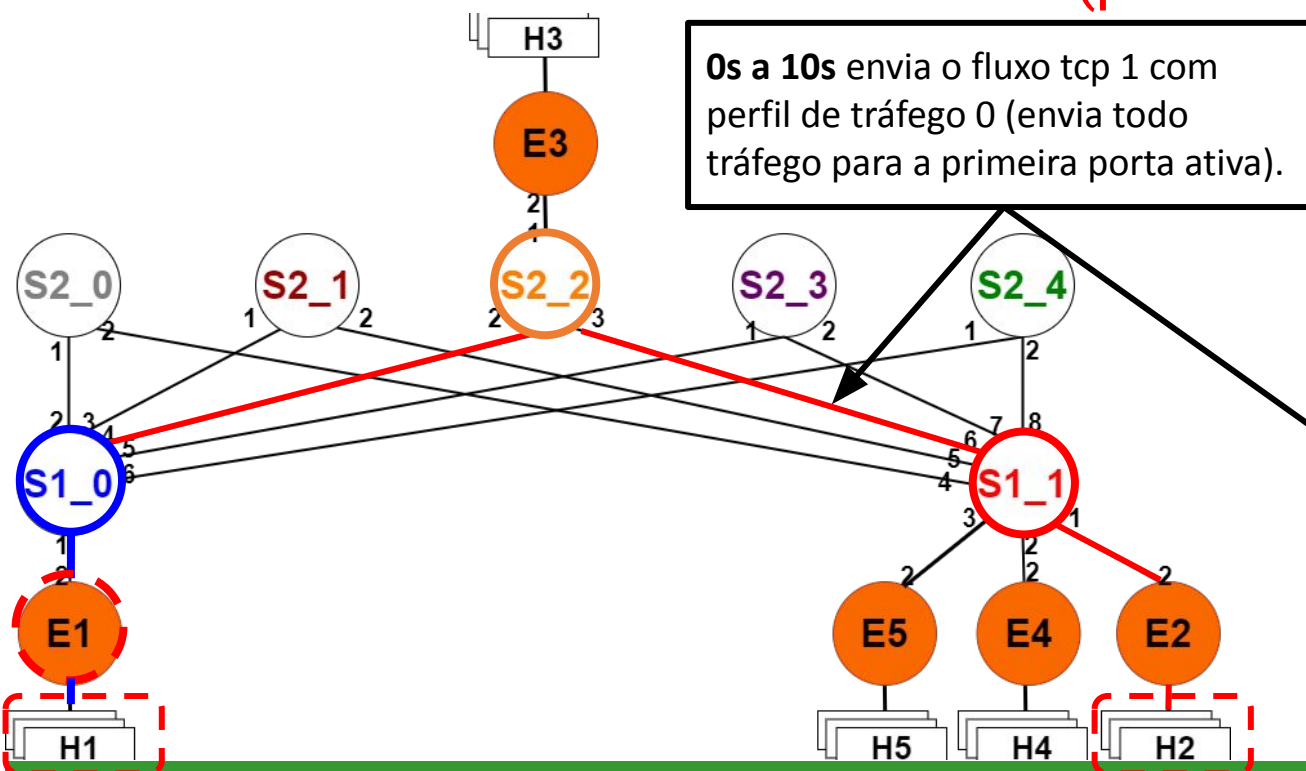
- Teste 3: Múltiplos fluxos e concorrência
 - Fluxo TCP1 entre H1 e H2 (período de 0s a 10s);

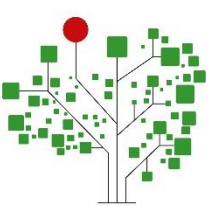




Avaliação Experimental

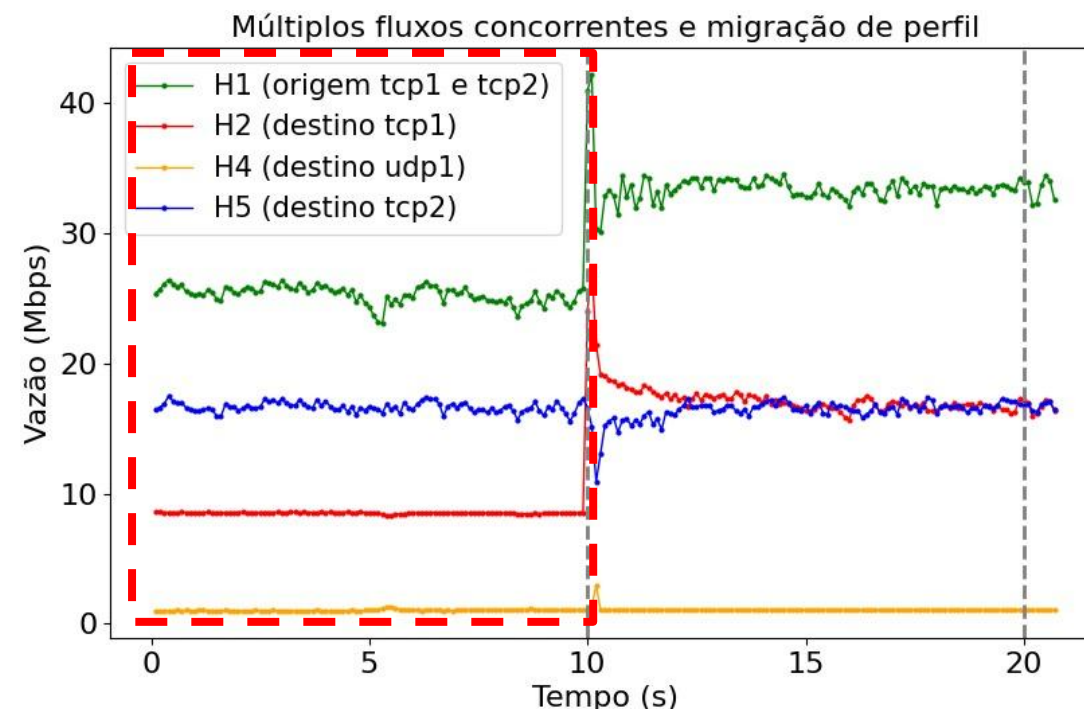
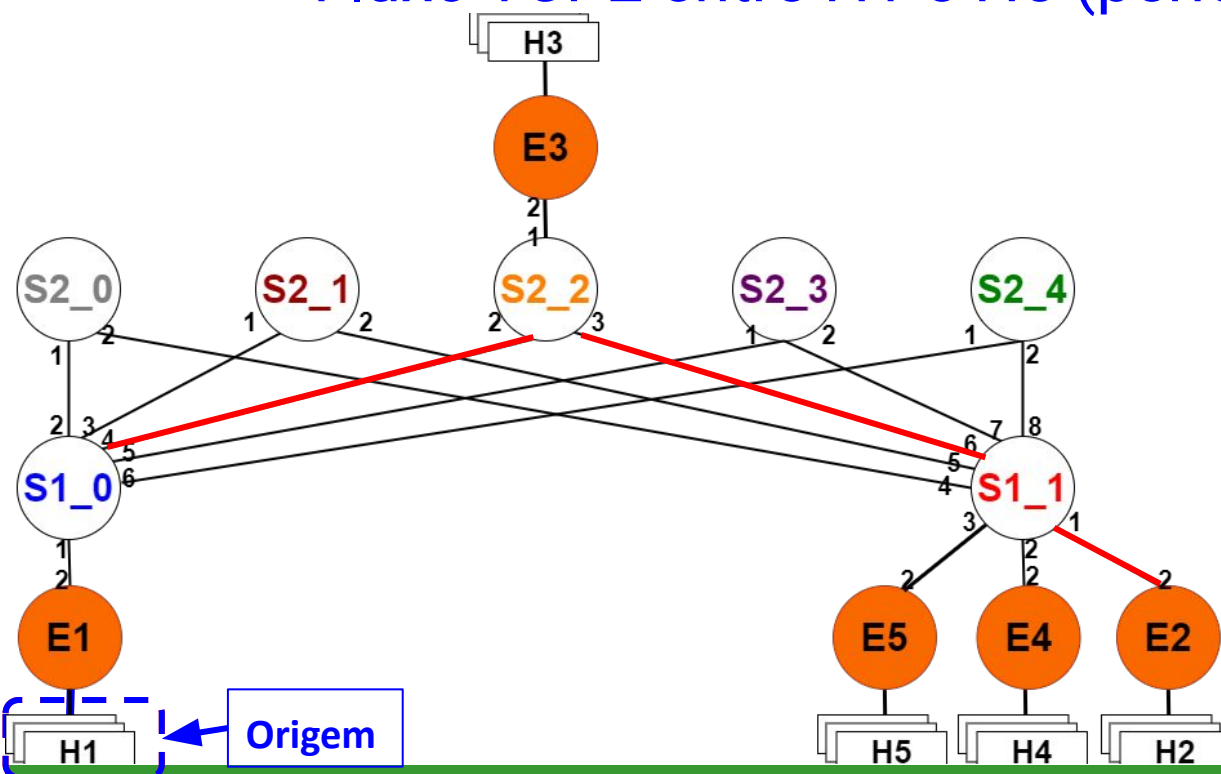
- Teste 3: Múltiplos fluxos e concorrência
 - Fluxo TCP1 entre H1 e H2 (período de 0s a 10s);





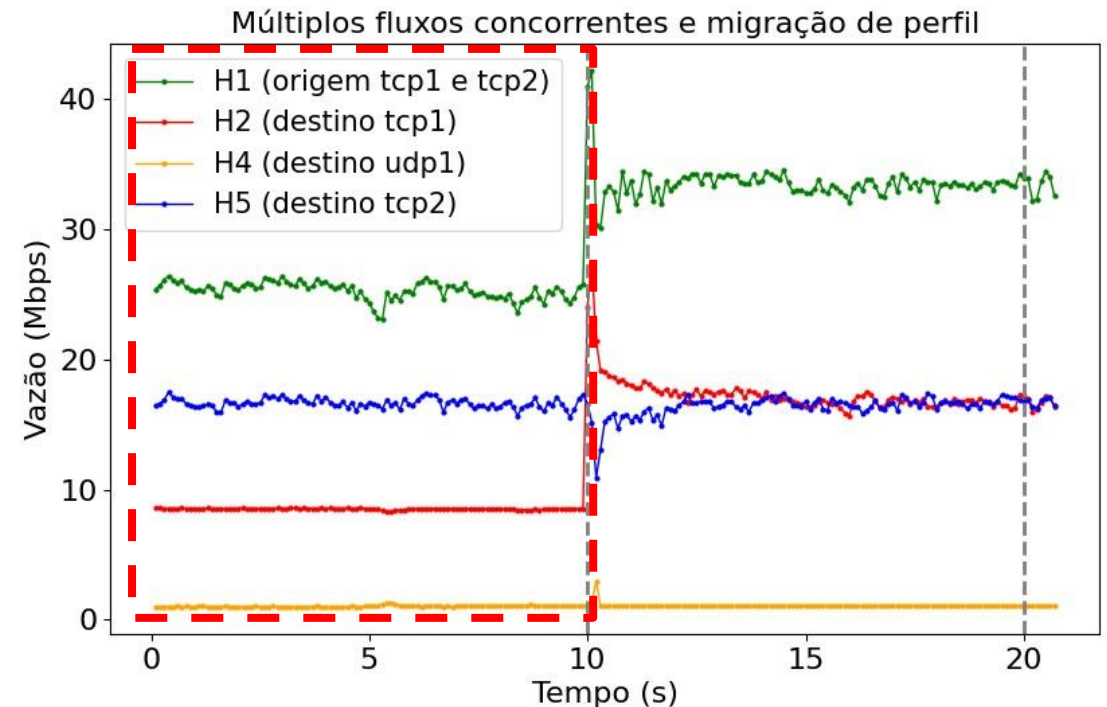
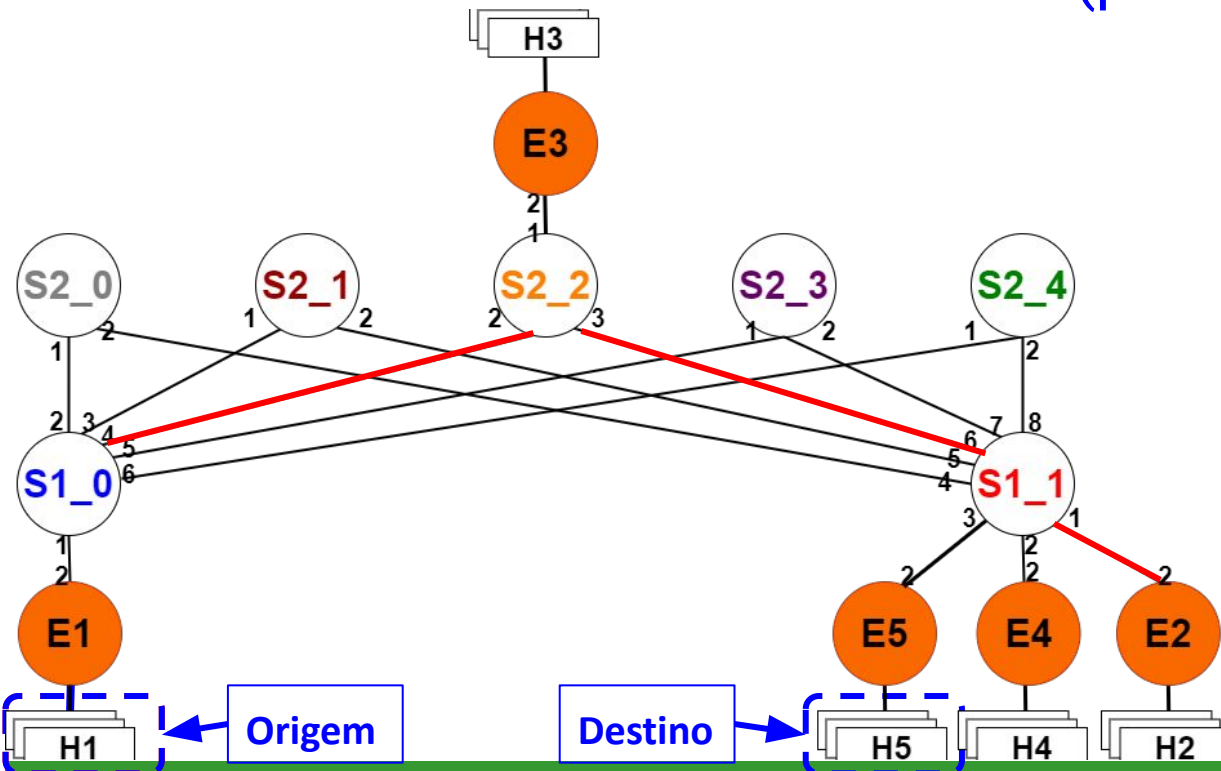
Avaliação Experimental

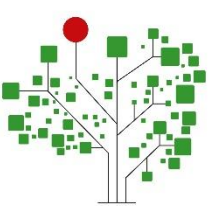
- Teste 3: Múltiplos fluxos e concorrência
 - Fluxo TCP2 entre H1 e H5 (período de 0s a 10s);



Avaliação Experimental

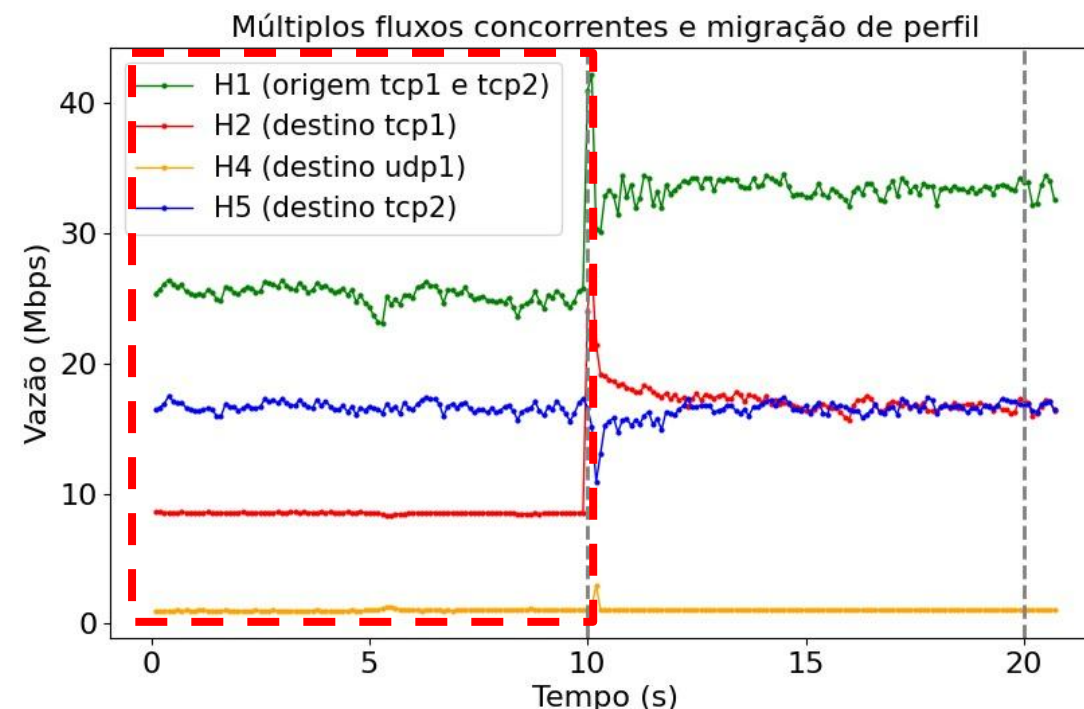
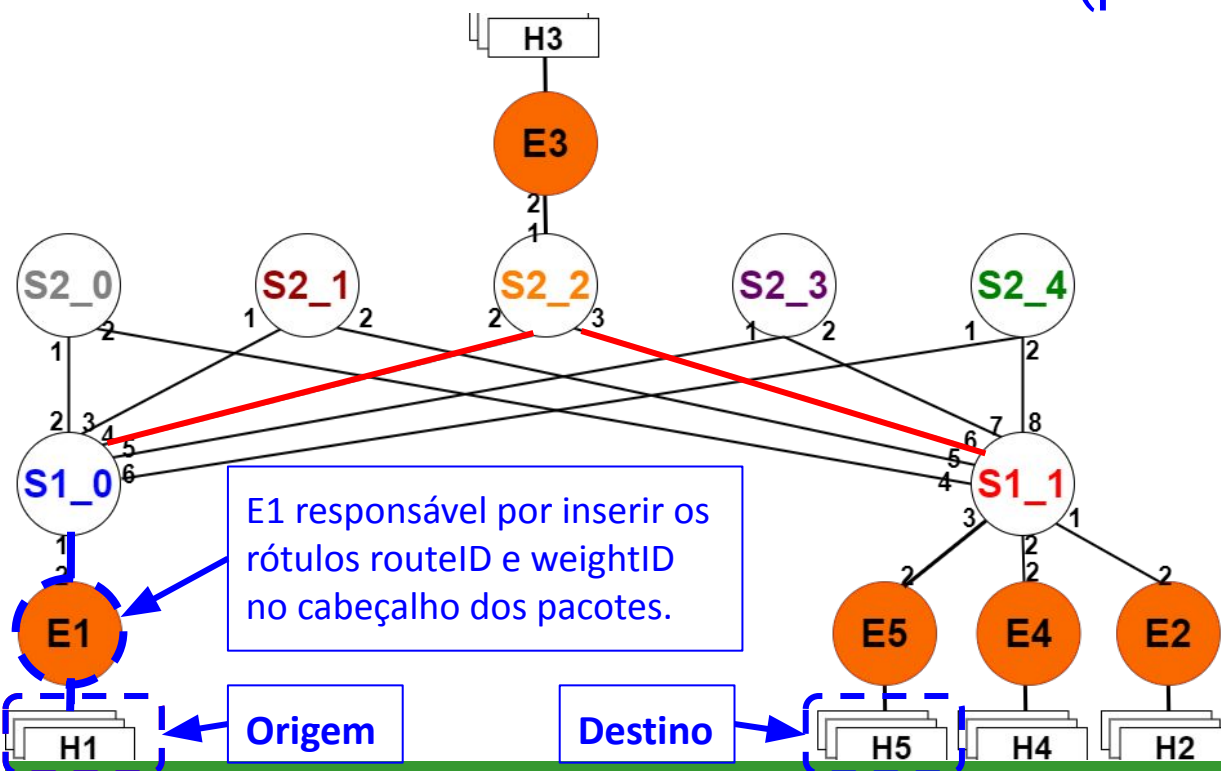
- Teste 3: Múltiplos fluxos e concorrência
 - Fluxo TCP2 entre H1 e H5 (período de 0s a 10s);

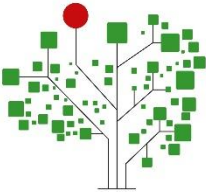




Avaliação Experimental

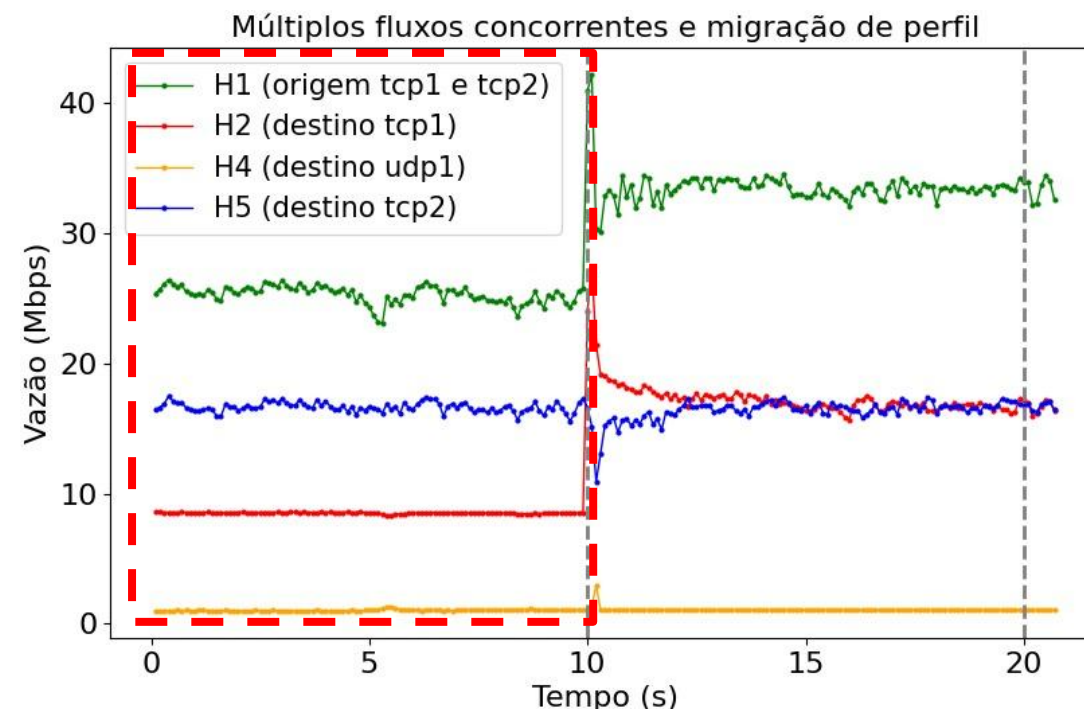
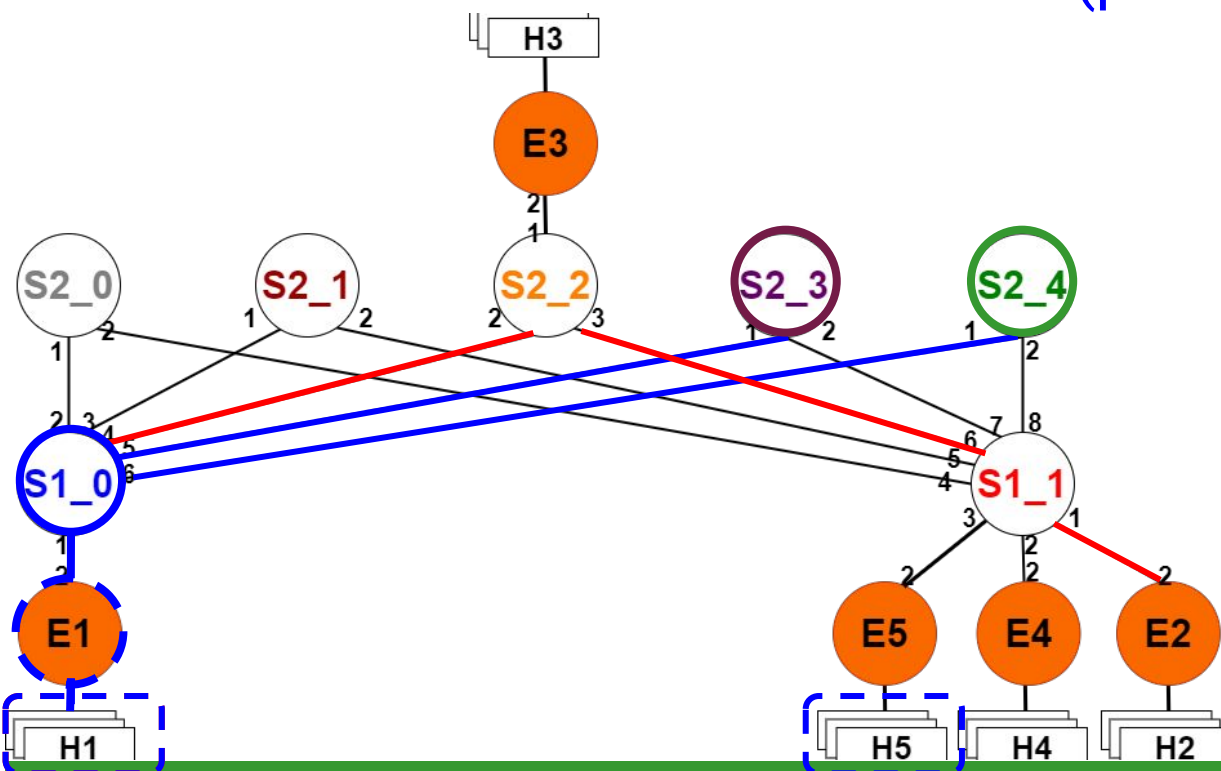
- Teste 3: Múltiplos fluxos e concorrência
 - Fluxo TCP2 entre H1 e H5 (período de 0s a 10s);

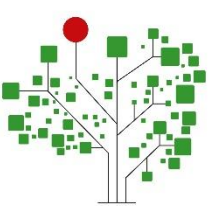




Avaliação Experimental

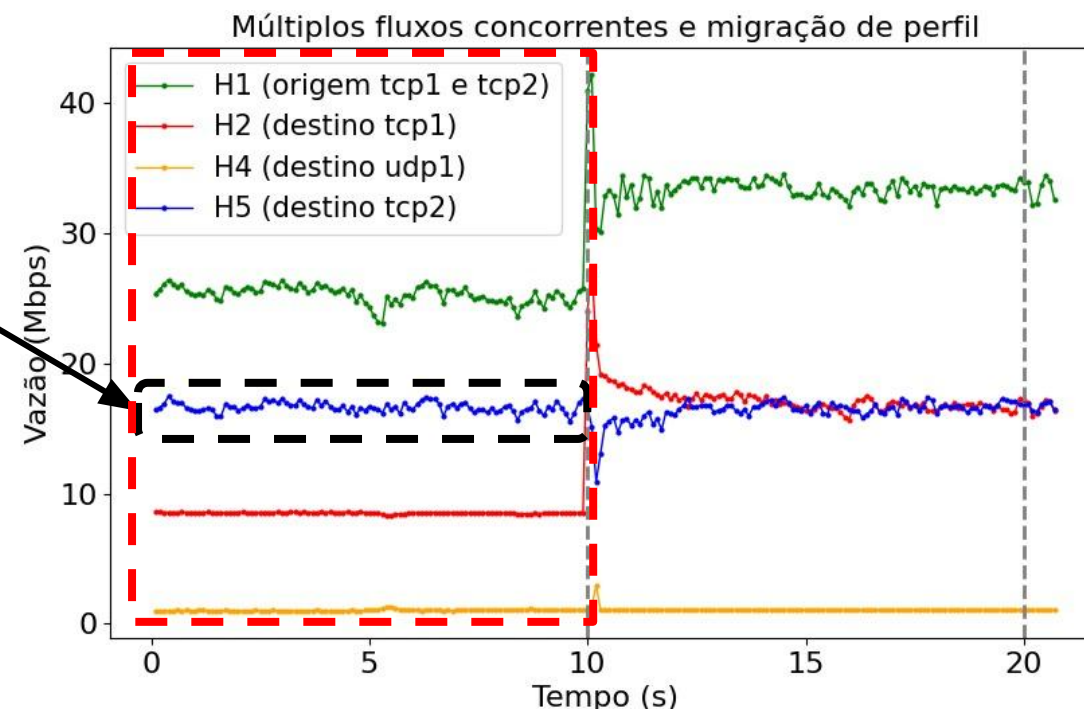
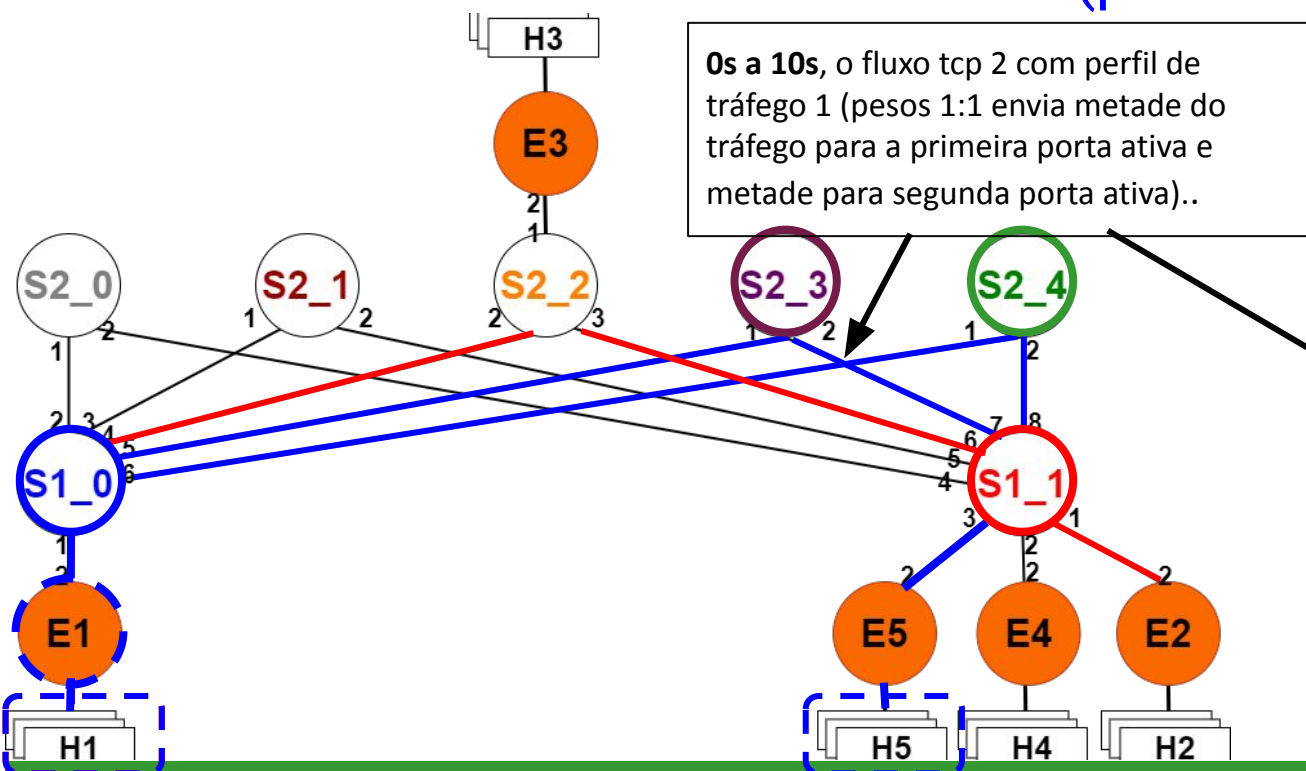
- Teste 3: Múltiplos fluxos e concorrência
 - Fluxo TCP2 entre H1 e H5 (período de 0s a 10s);

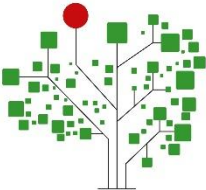




Avaliação Experimental

- Teste 3: Múltiplos fluxos e concorrência
 - Fluxo TCP2 entre H1 e H5 (período de 0s a 10s);

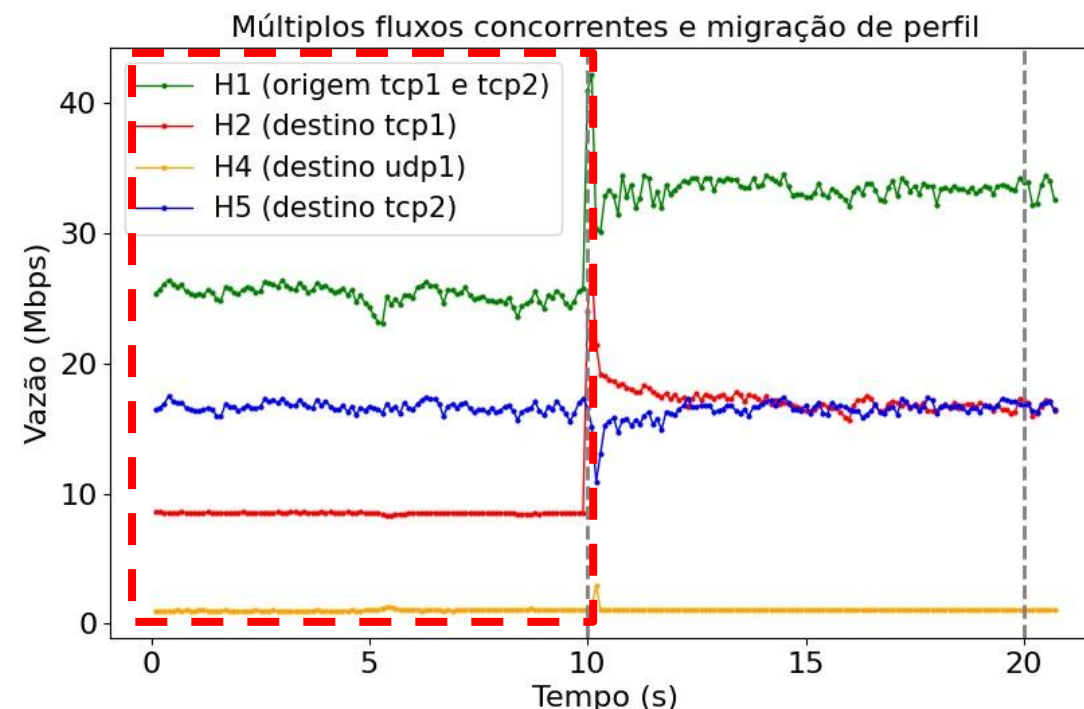
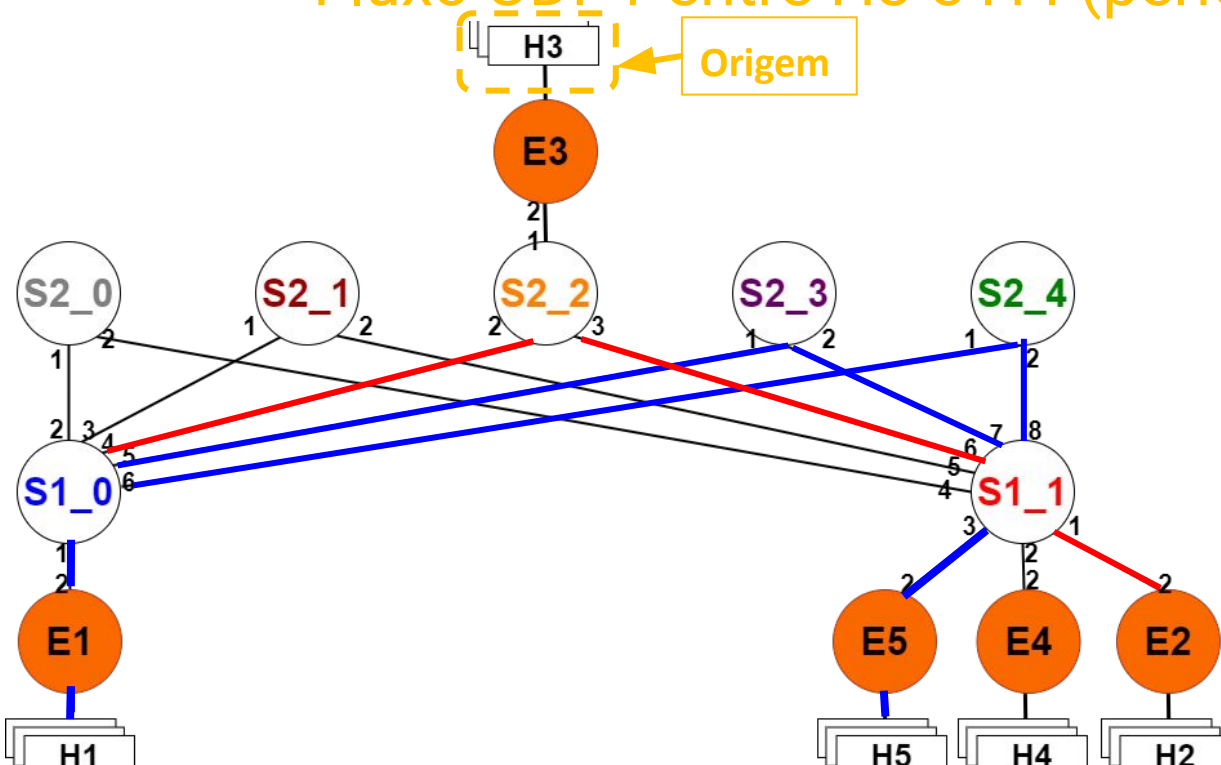


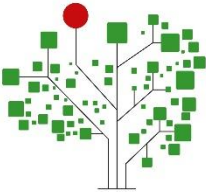


Avaliação Experimental

- Teste 3: Múltiplos fluxos e concorrência

- Fluxo UDP1 entre H3 e H4 (período de 0s a 10s);

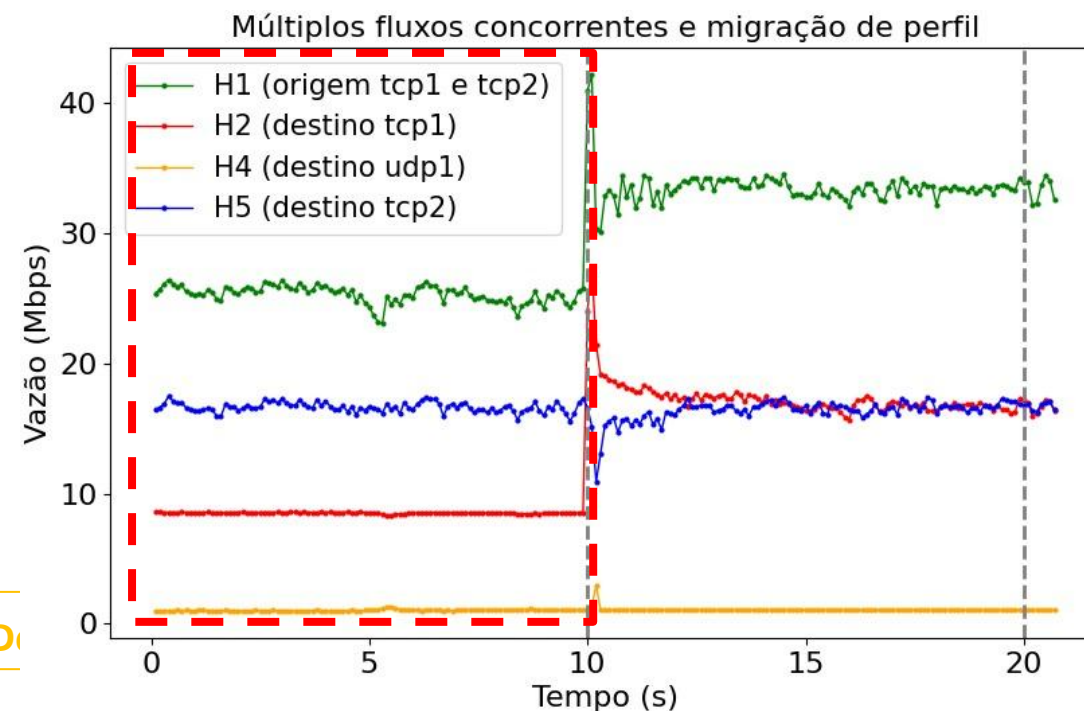
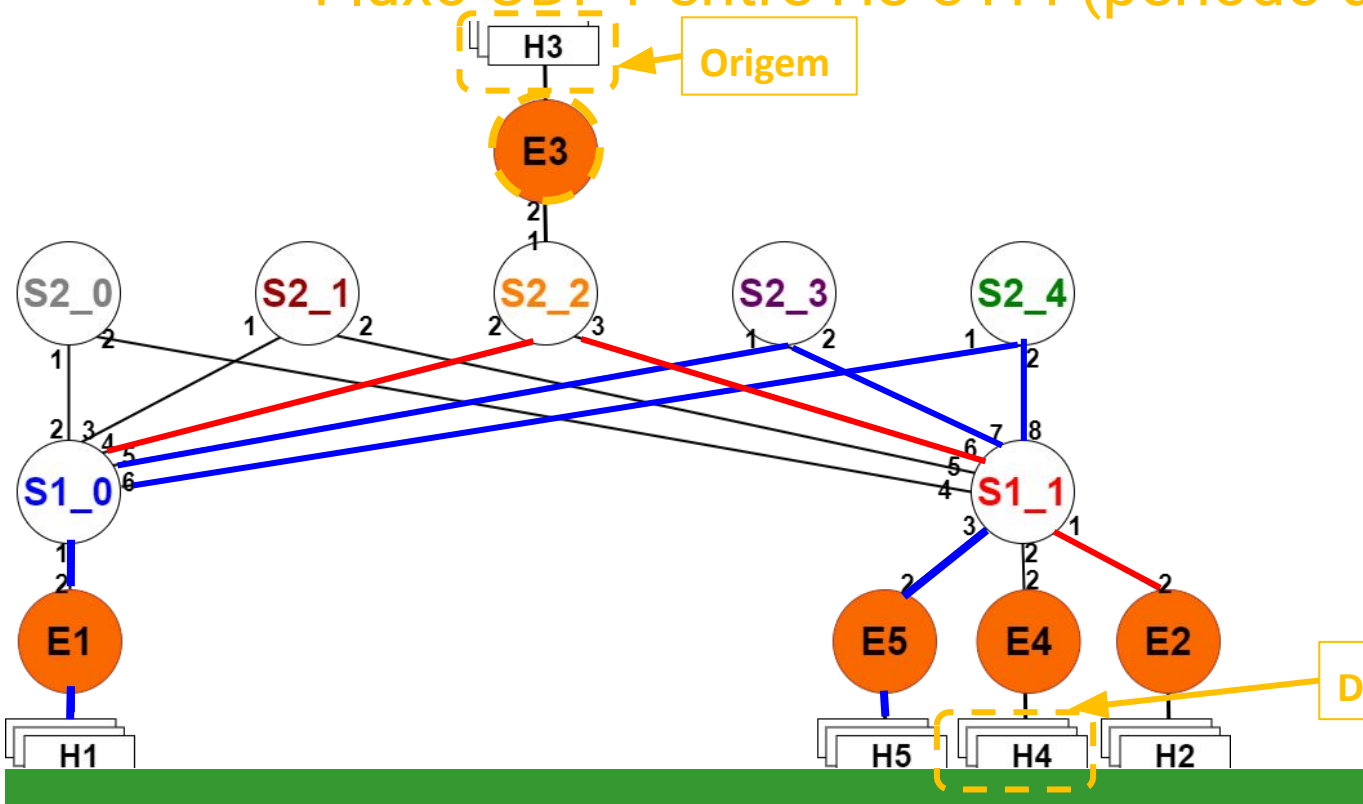




Avaliação Experimental

- Teste 3: Múltiplos fluxos e concorrência

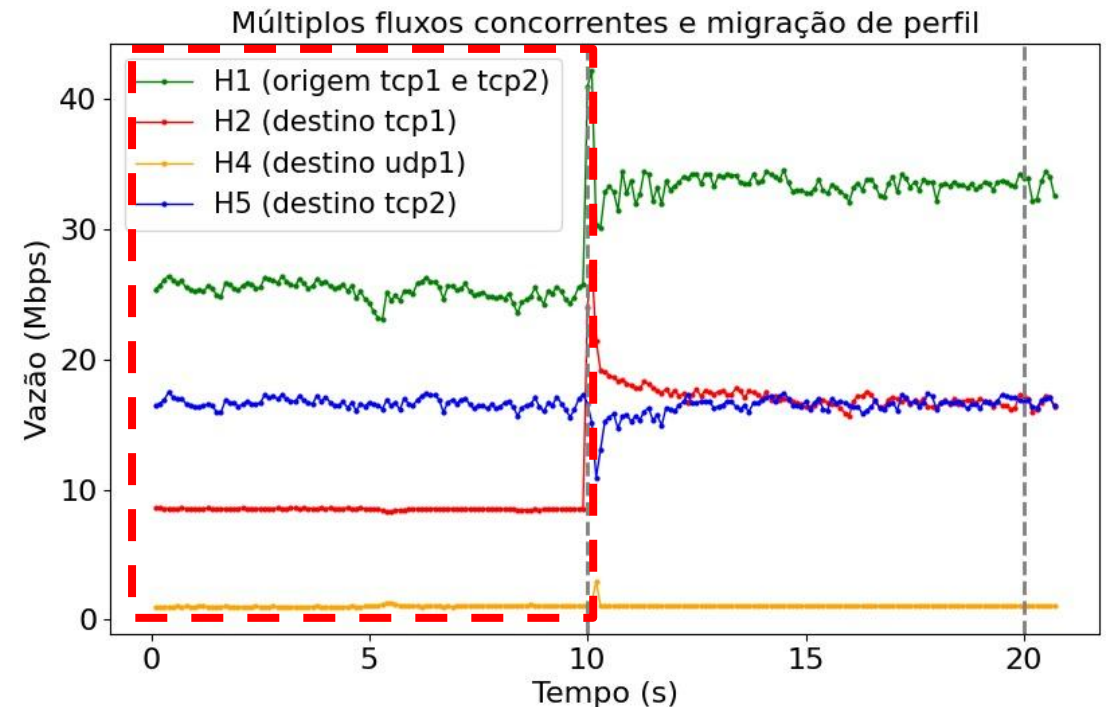
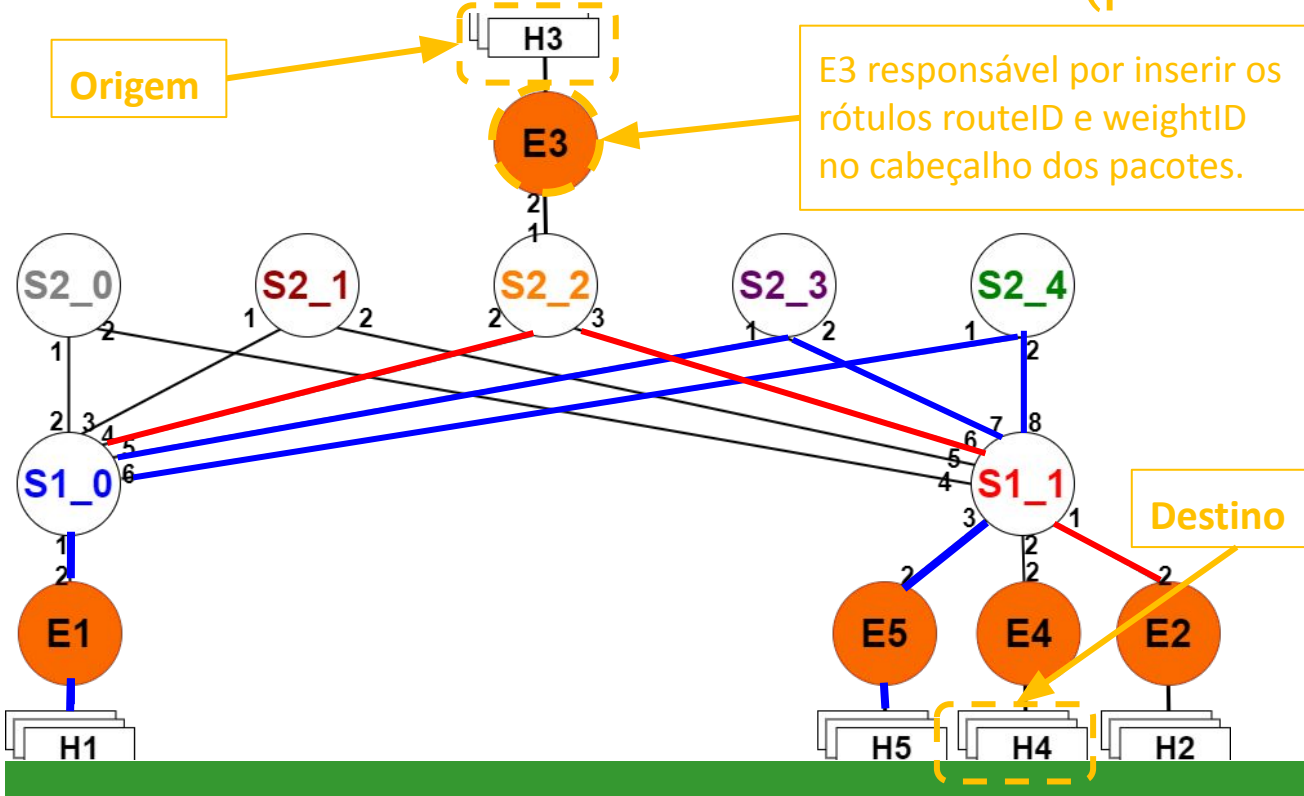
- Fluxo UDP1 entre H3 e H4 (período de 0s a 10s);

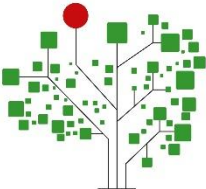


Avaliação Experimental

• Teste 3: Múltiplos fluxos e concorrência

- Fluxo UDP1 entre H3 e H4 (período de 0s a 10s);

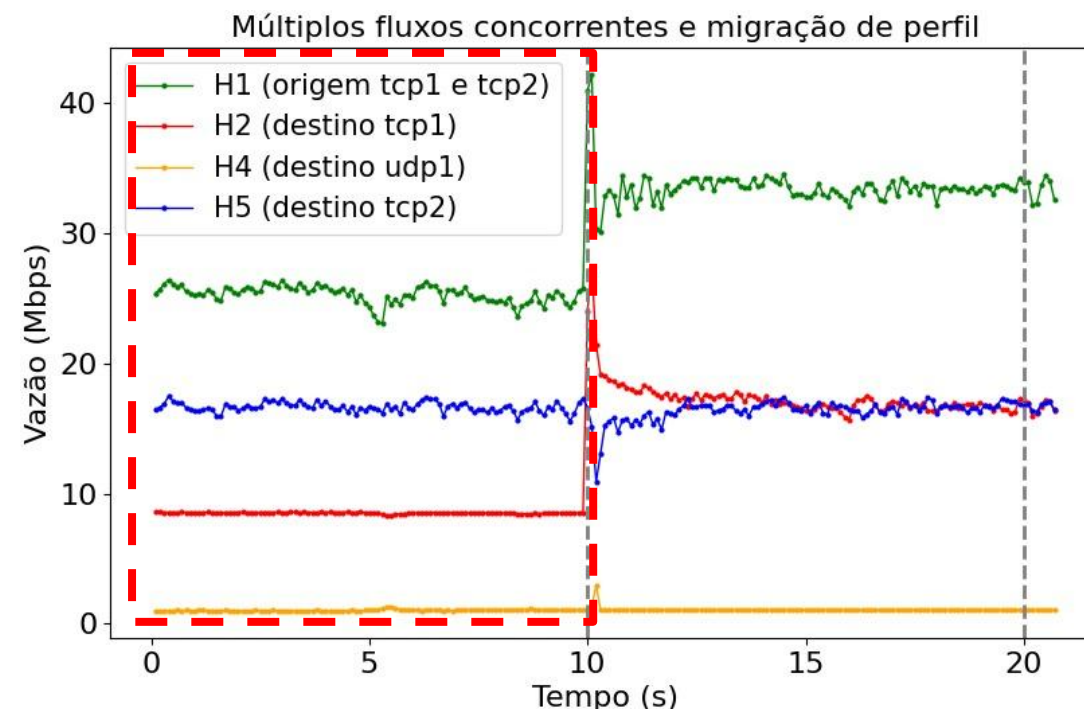
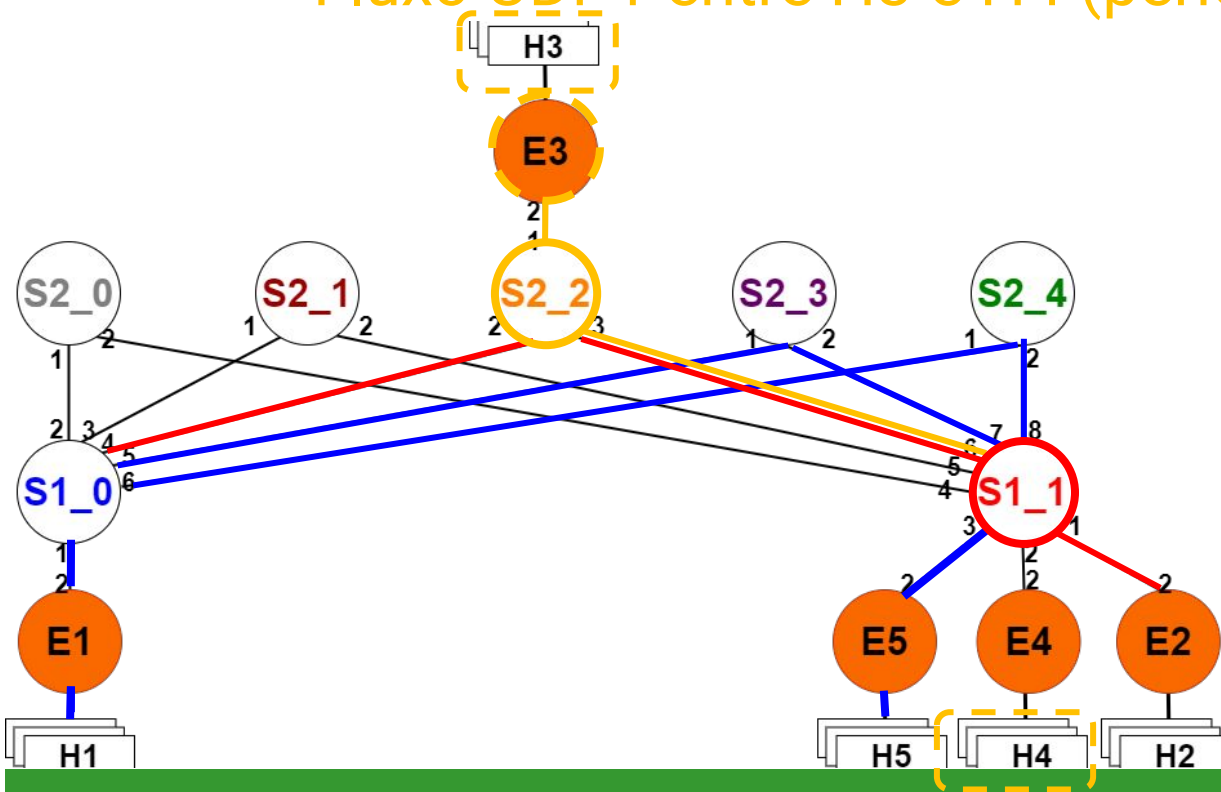


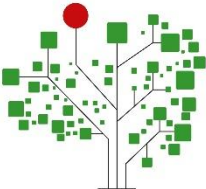


Avaliação Experimental

- Teste 3: Múltiplos fluxos e concorrência

- Fluxo UDP1 entre H3 e H4 (período de 0s a 10s);

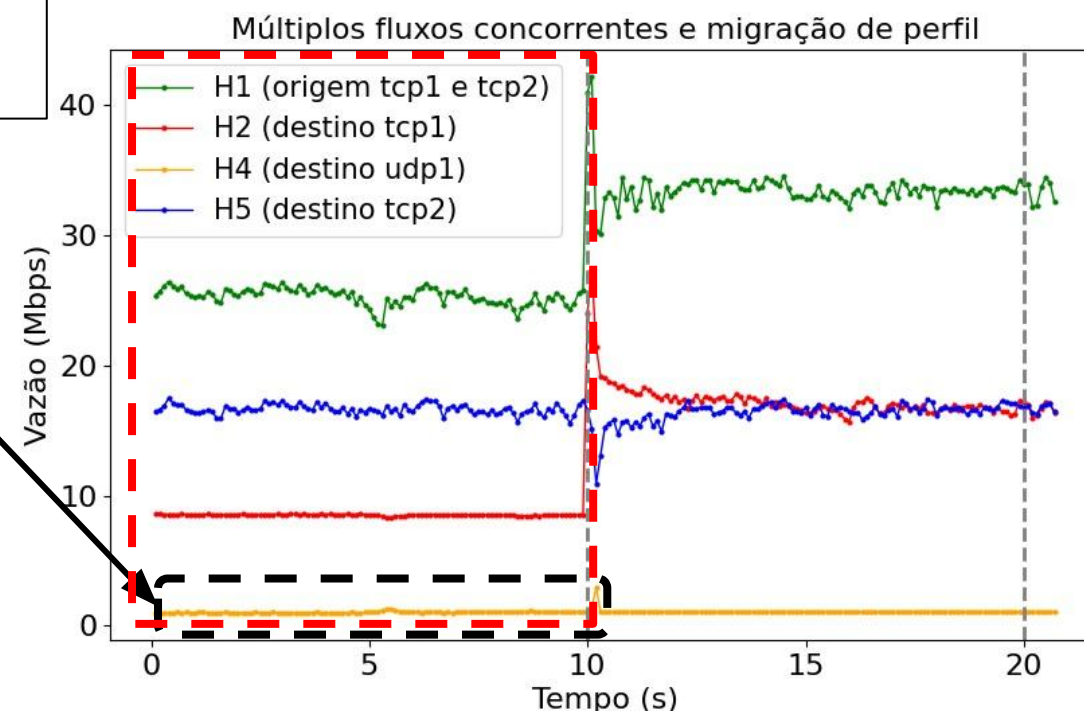
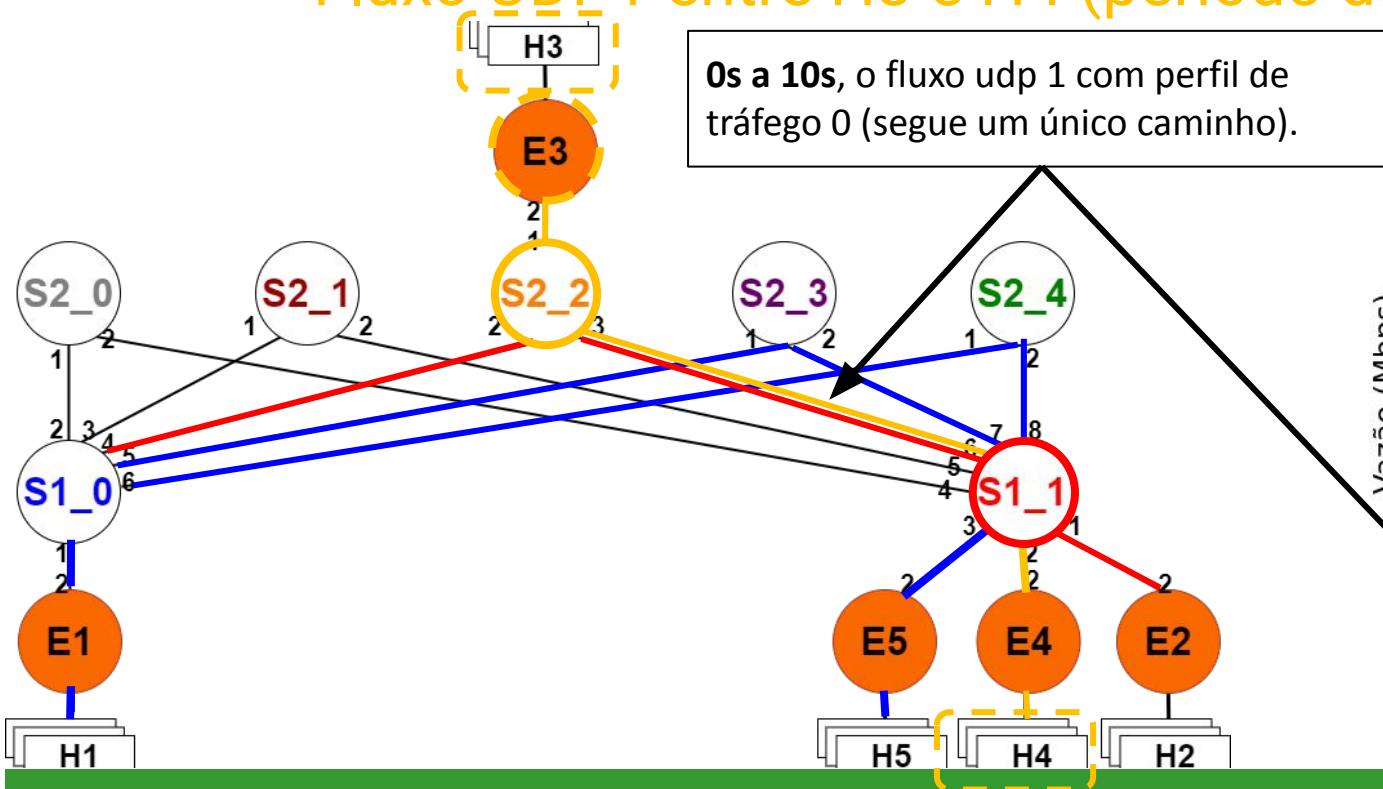


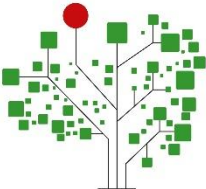


Avaliação Experimental

- Teste 3: Múltiplos fluxos e concorrência

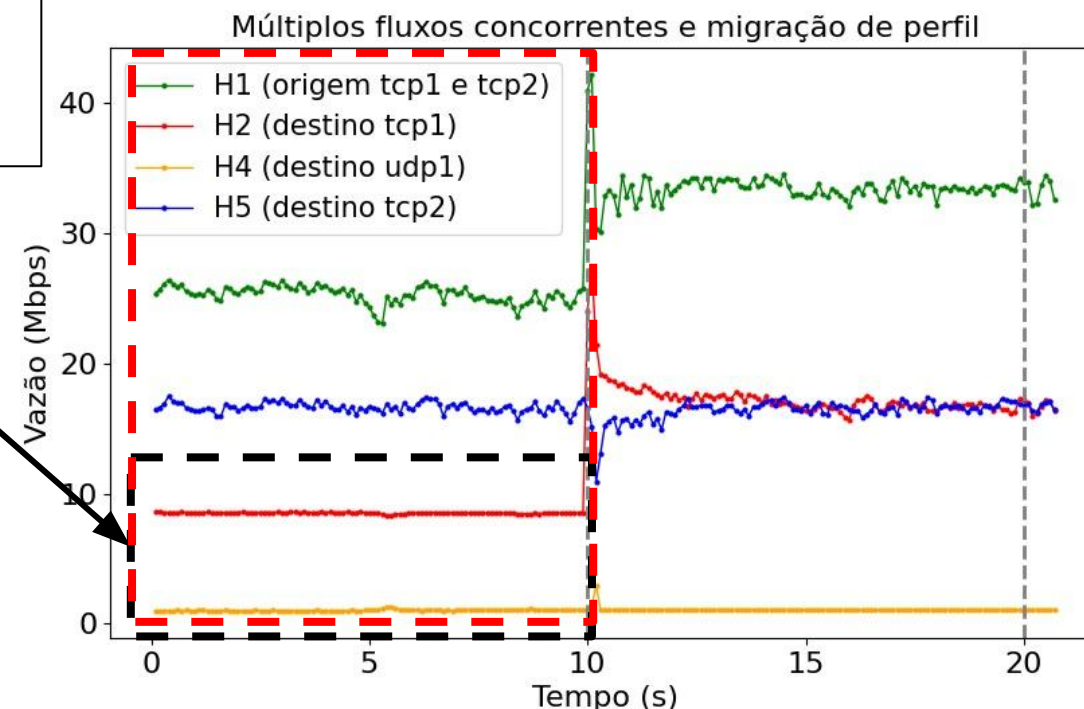
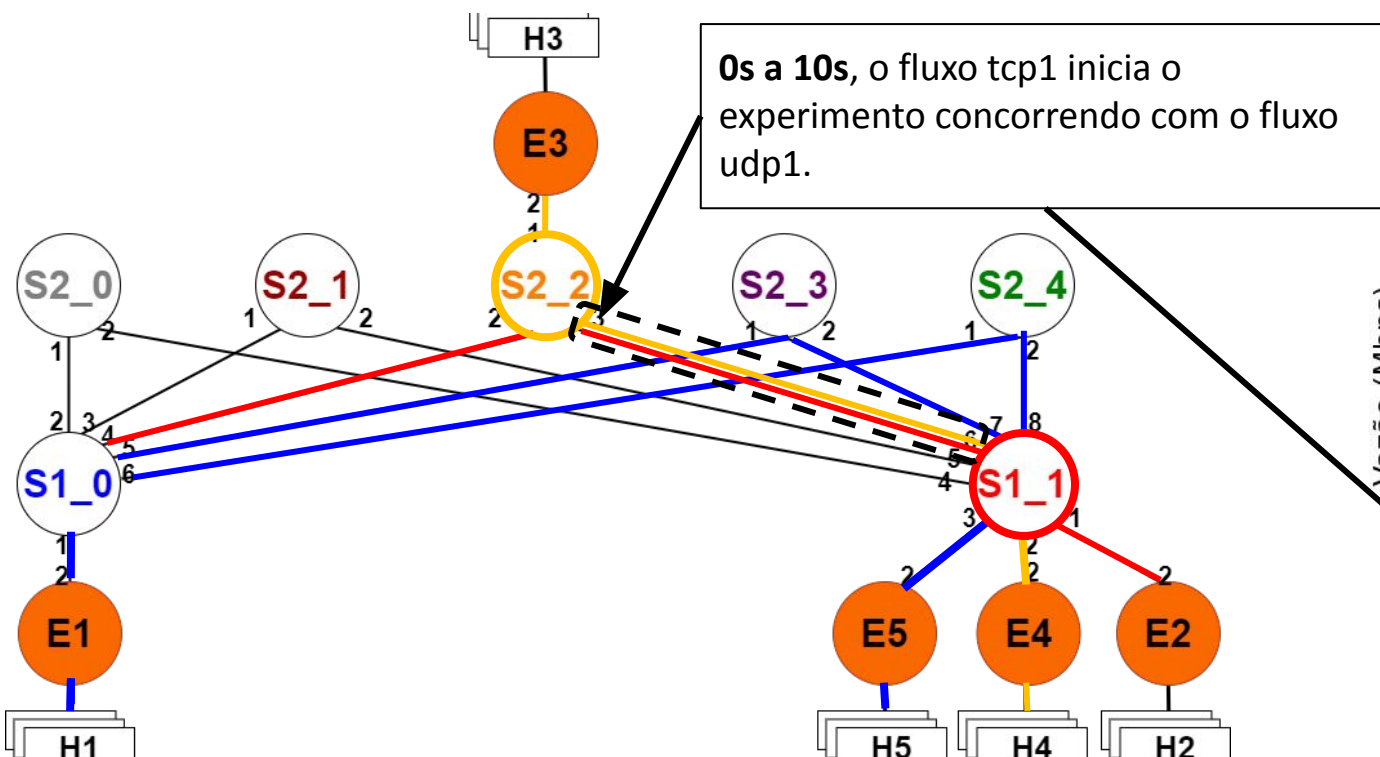
- Fluxo UDP1 entre H3 e H4 (período de 0s a 10s);

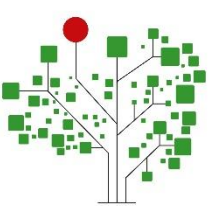




Avaliação Experimental

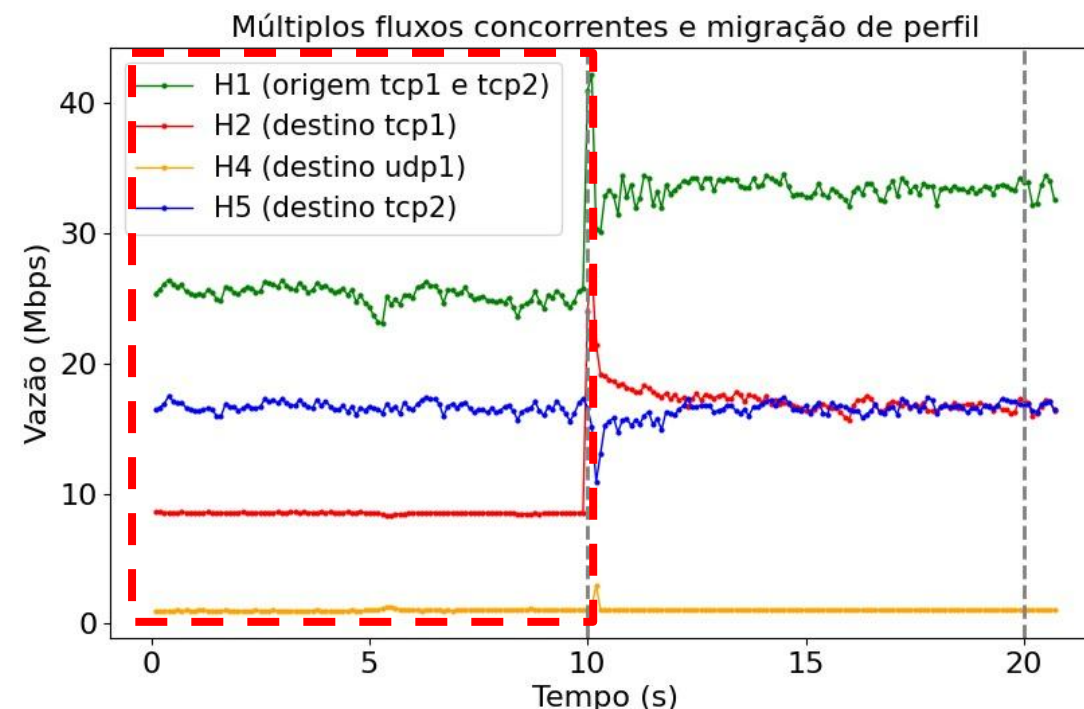
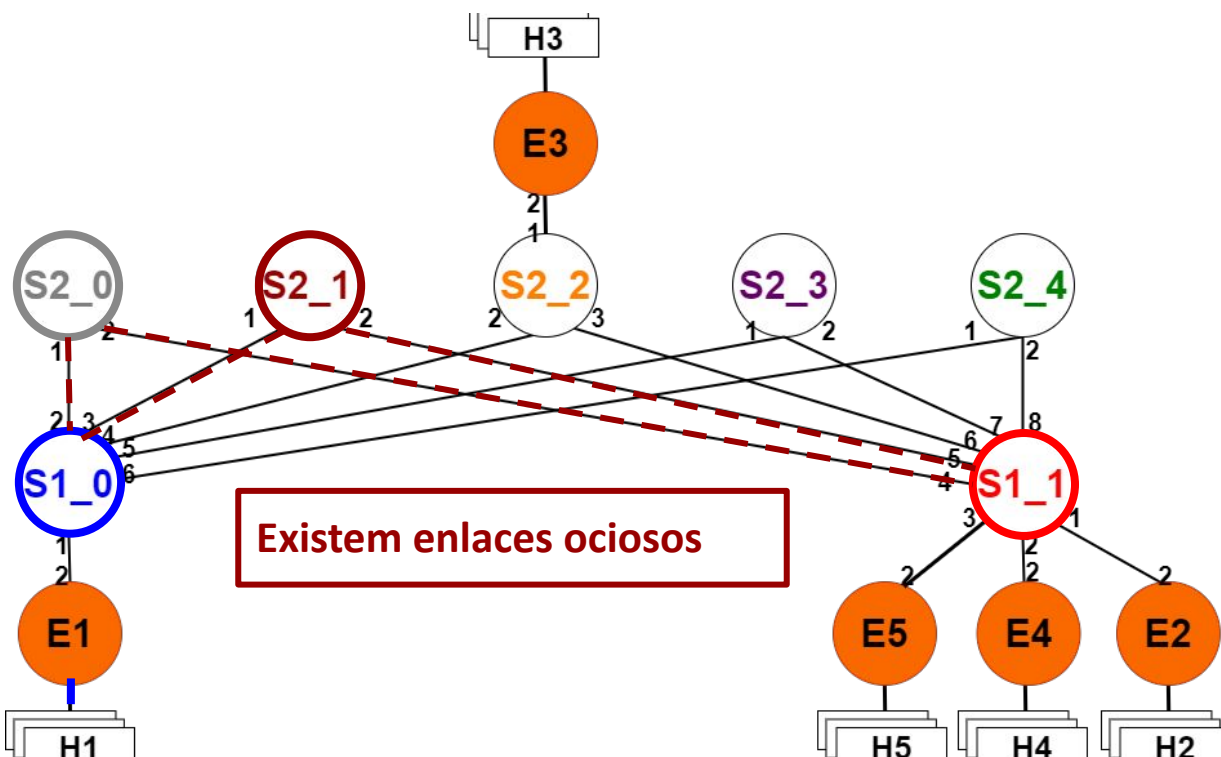
- Teste 3: Múltiplos fluxos e concorrência
 - Fluxo TCP1 concorrendo com o fluxo UDP1 (período de 0s a 10s);

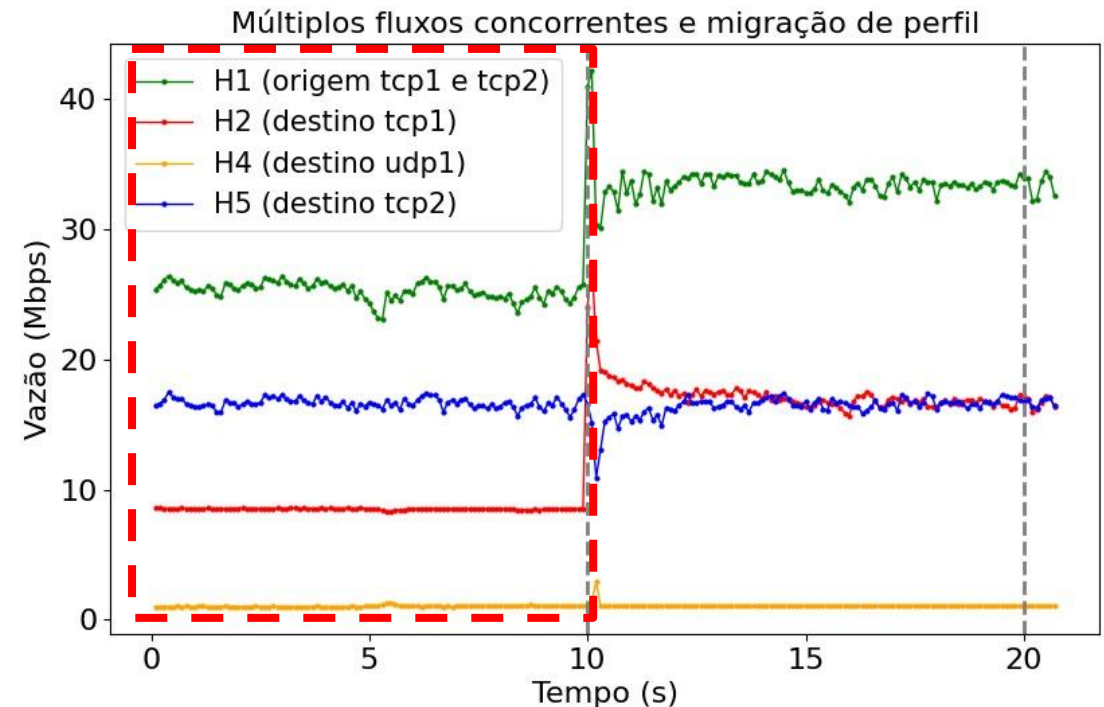


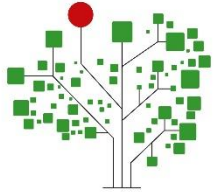


Avaliação Experimental

- Teste 3: Múltiplos fluxos e concorrência
 - Fluxo TCP1 concorrendo com o fluxo UDP1;

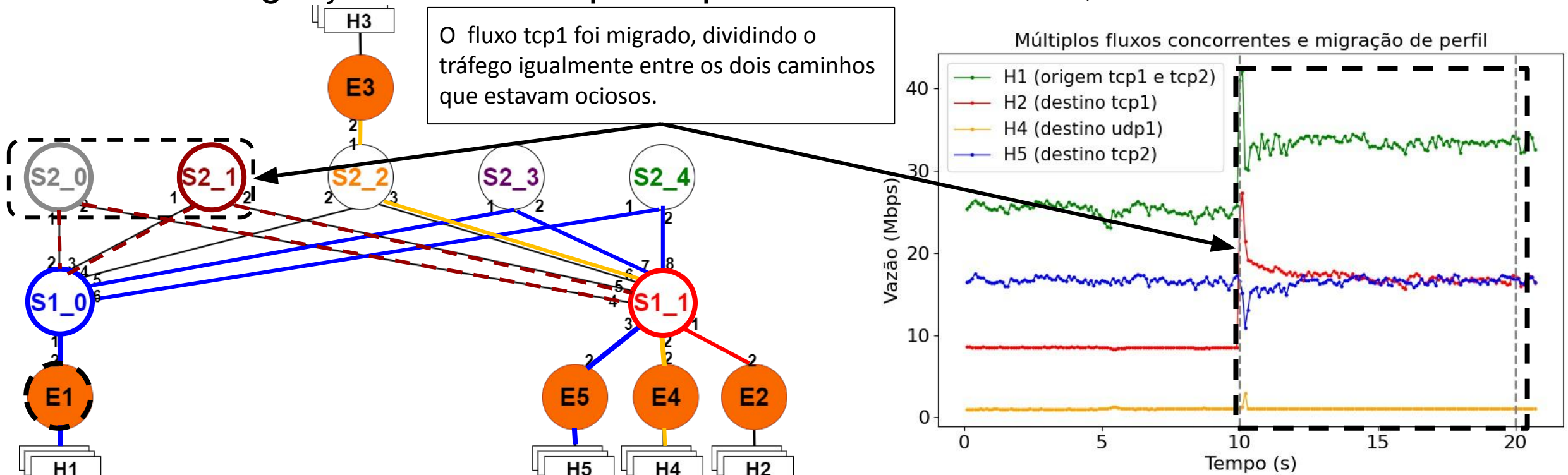


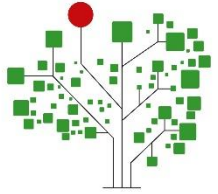




Avaliação Experimental

- Teste 3: Múltiplos fluxos e concorrência
 - Migração do fluxo tcp1 no período de 10s a 20s;

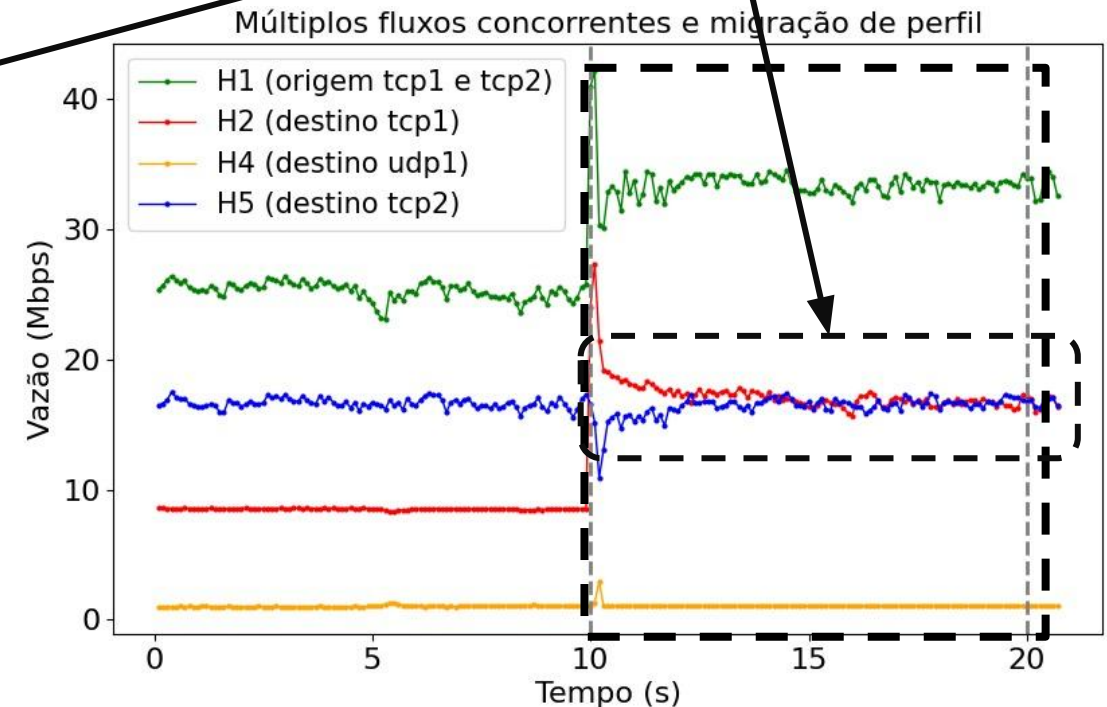
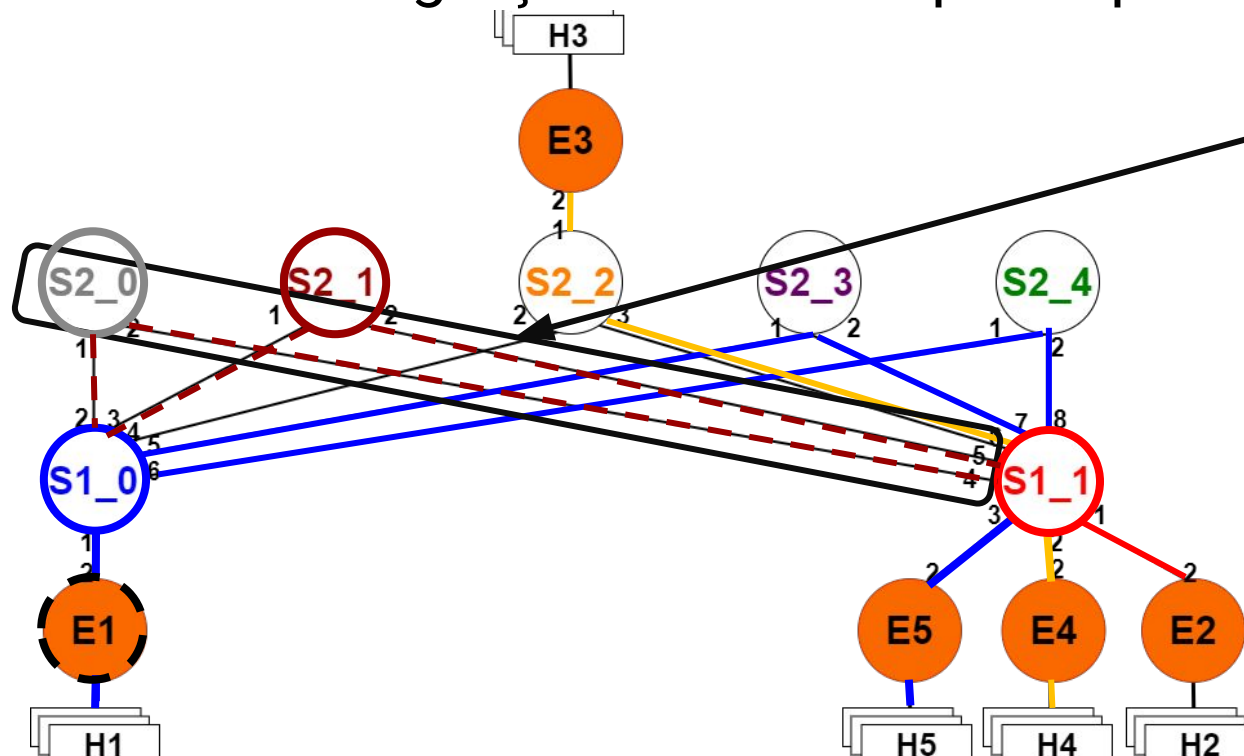


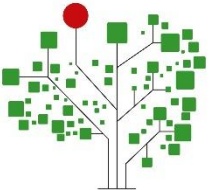


Avaliação Experimental

- Teste 3: Múltiplos fluxos e concorrência
 - Migração do fluxo tcp1 no período de 10s a 20s;

Aumento da vazão atingida pelo fluxo tcp1 após a migração (linha vermelha);

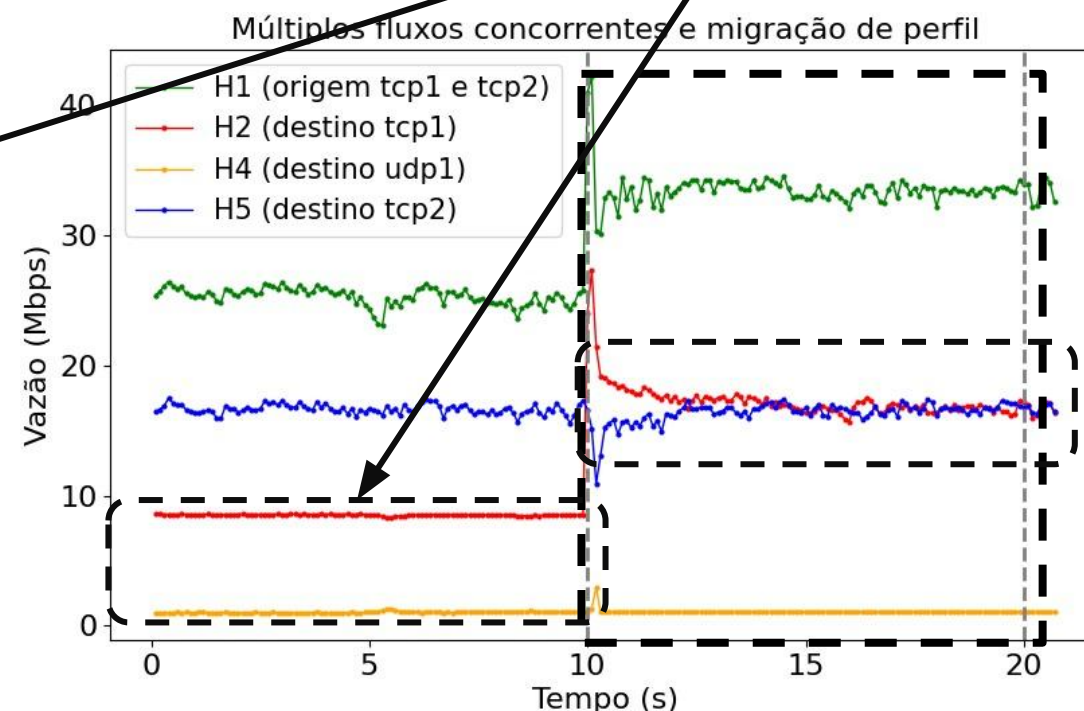
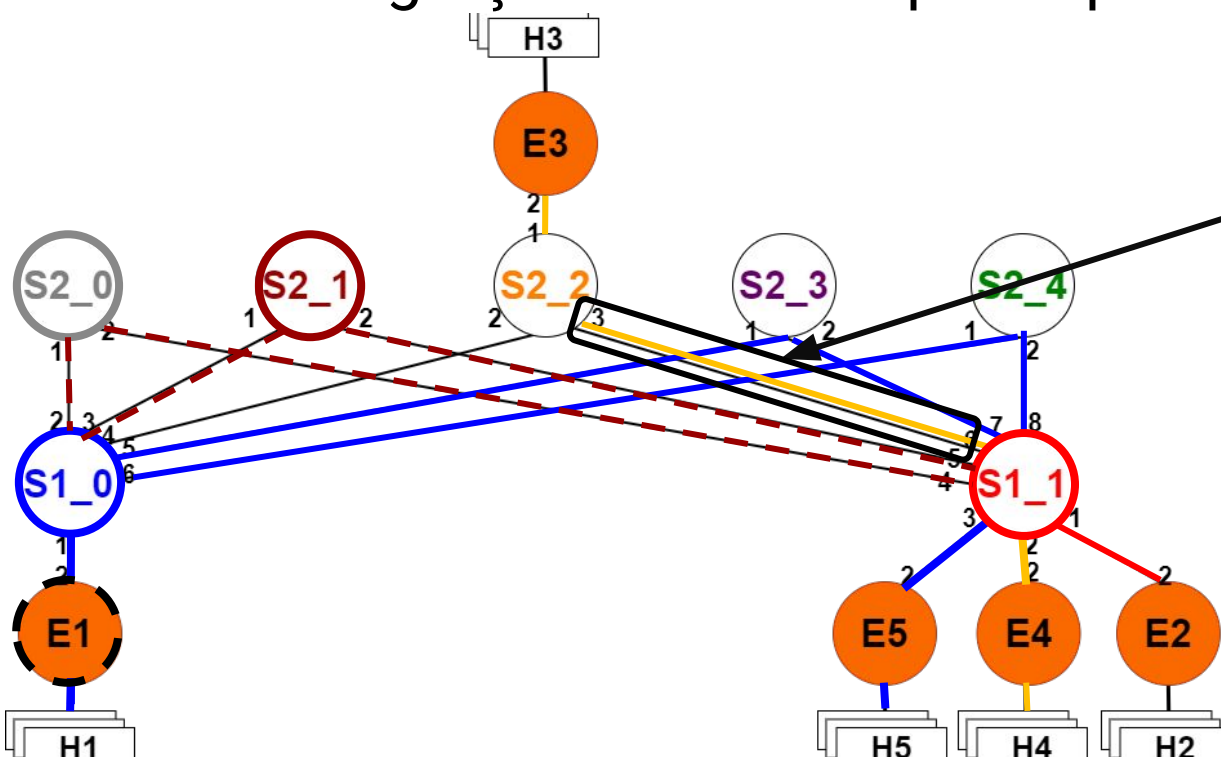


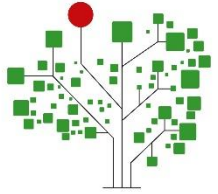


Avaliação Experimental

- Teste 3: Múltiplos fluxos e concorrência
 - Migração do fluxo tcp1 no período de 10s a 20s;

Elimina a concorrência com o fluxo udp1 e explorando dois caminhos;

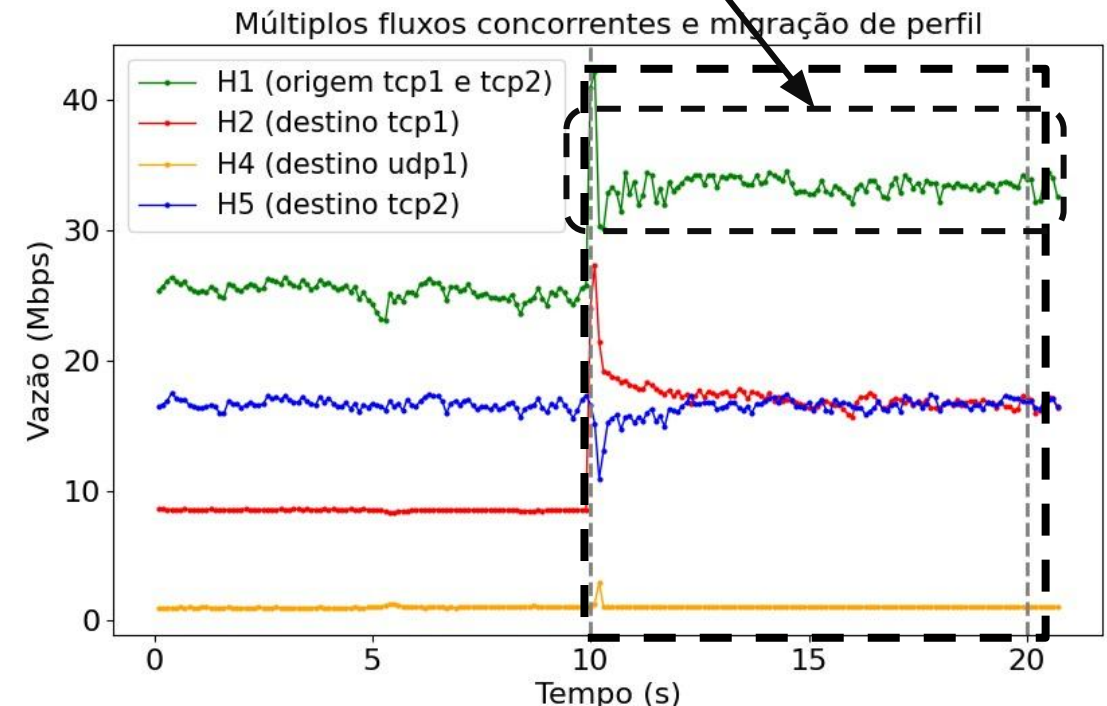
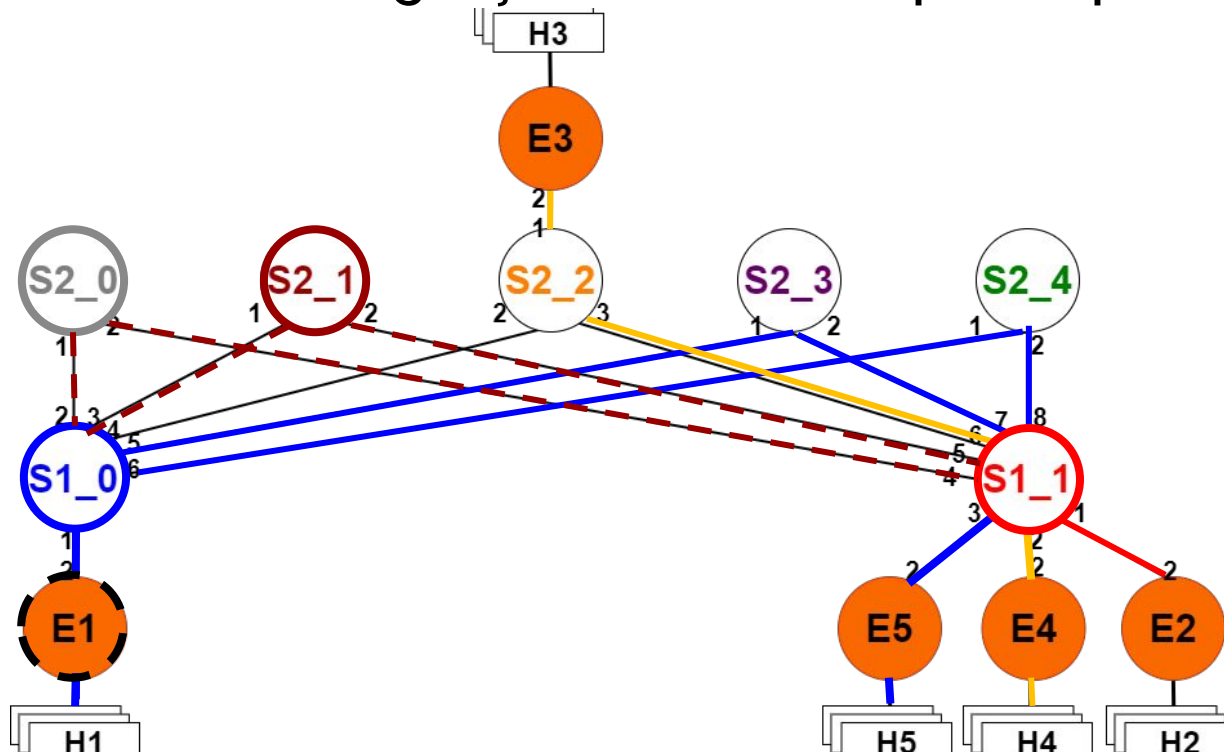




Avaliação Experimental

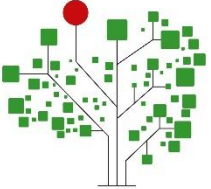
- Teste 3: Múltiplos fluxos e concorrência
 - Migração do fluxo tcp1 no período de 10s a 20s;

Aumento da vazão agregada no host H1 devido à exploração de multicaminhos (linha verde);



Conclusão

- O MTS-PolKA é uma abordagem inovadora para seleção de perfil de divisão de tráfego na origem, baseada em proporção de pesos em uma árvore de distribuição de tráfego.
- Diferentemente de abordagens tradicionais como ECMP e WCMP, o MTS-PolKA consegue adaptar a distribuição de tráfego de forma ágil em todos os switches do caminho, com pequenas modificações no cabeçalho do pacote na borda, sem alterações no núcleo da rede.



PPComp

Mestrado Profissional em
Computação Aplicada

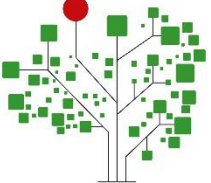


INSTITUTO FEDERAL

Espírito Santo
Campus Serra

Trabalhos Futuros

- Extensão da avaliação de desempenho e escalabilidade da proposta em comparação com trabalhos relacionados em diferentes cenários.
- Análise da sobrecarga de memória e banda do método.
- Experimentos de reordenação de pacotes.
- Implementação em switches programáveis Tofino®.



PPComp

Mestrado Profissional em
Computação Aplicada



INSTITUTO FEDERAL
Espírito Santo
Campus Serra

OBRIGADO!