

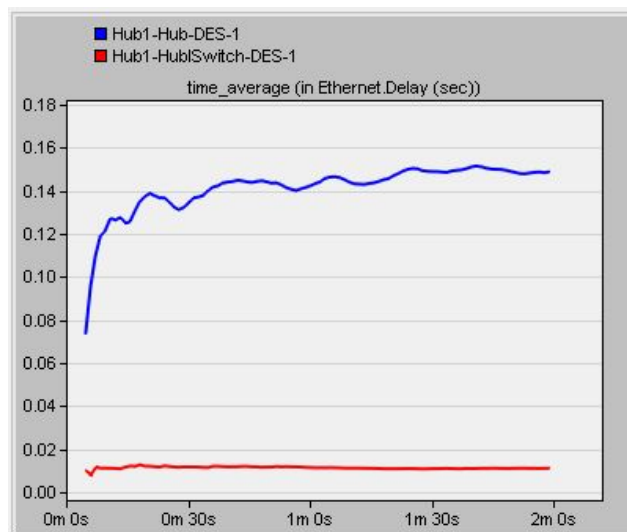
## PREGUNTES

### 1. Expliqueu si hi ha alguna diferència entre el escenari Hub i el escenari HubISwitch, i el motiu d'aquesta.

En l'escenari Hub, només disposem d'una xarxa, en la qual hi ha un Hub que connecta tots els nodes. Aquest Hub, rep la informació d'algun dels nodes de la xarxa local i la transmet a la resta.

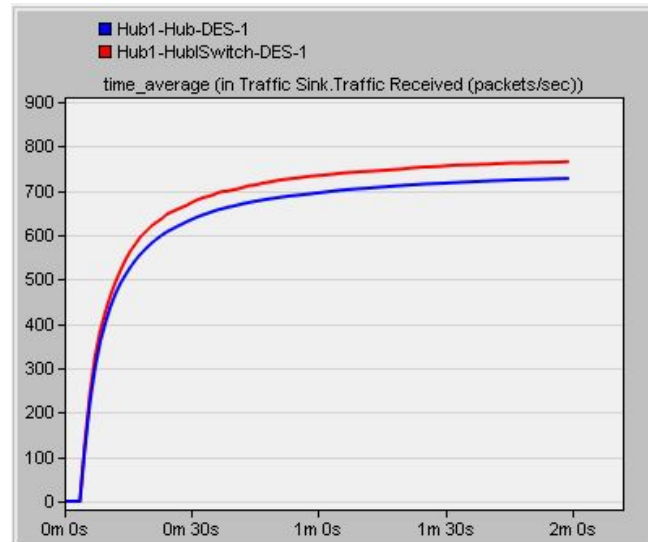
En el segon escenari, disposem de dues xarxes, que com a l'escenari anterior tenen cada una un Hub propi, i estan connectades a través d'un Switch. Encarregat de rebre la informació dels Hubs i, si es requereix, de transmetre-la a l'altre Hub per a que la envii a tots els nodes connectats a ell, un d'ells, el destinatari final.

Ethernet → Delay(sec)



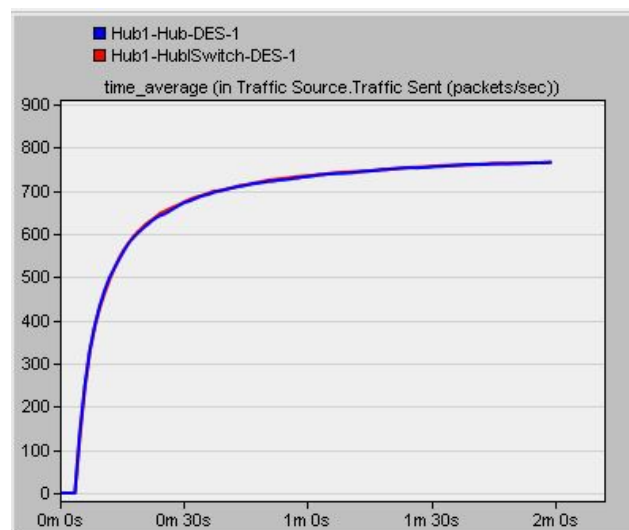
El delay mitjà de l'escenari Hub és aproximadament deu cops el de l'escenari HubISwitch, degut a les col·lisions que es produeixen quan dos nodes envien un missatge al mateix temps. Aquest missatge es retornat a l'emissor i es tornarà a enviar fins que sigui rebut sense que hi hagi cap col·lisió. En canvi, en el segon escenari com tenim dos Hub, i per tant menys nodes connectats a ells, el nombre de col·lisions disminueix considerablement, i en conseqüència el delay també ho fa.

### Traffic Sink → Received (packets/sec)



Degut a que el delay es menor en l'escenari HubSwitch, aquest rep més paquets per segon, tot en conseqüència de que la probabilitat de que es produeixin col·lisions és menor que en l'altre escenari.

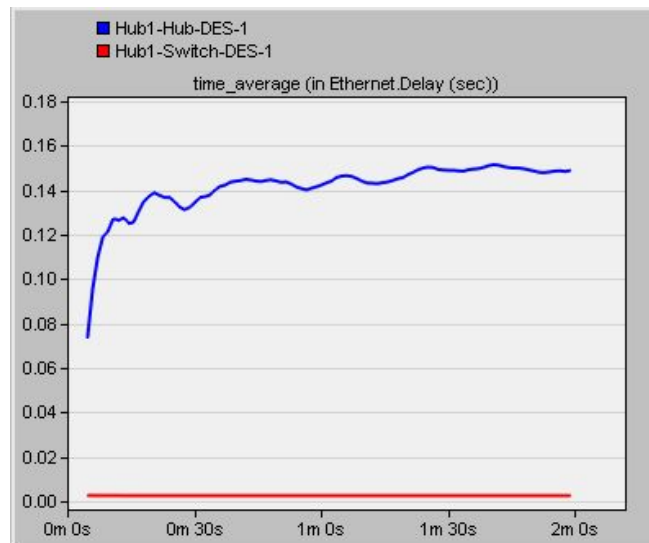
### Traffic Source → Traffic Sent (packets/sec)



La mitjana del temps que es tarda en enviar un paquet és la mateixa perquè un cop els Hubs han rebut el missatge, només han d'enviar-lo al destinatari, i ja no es podran produir col·lisions.

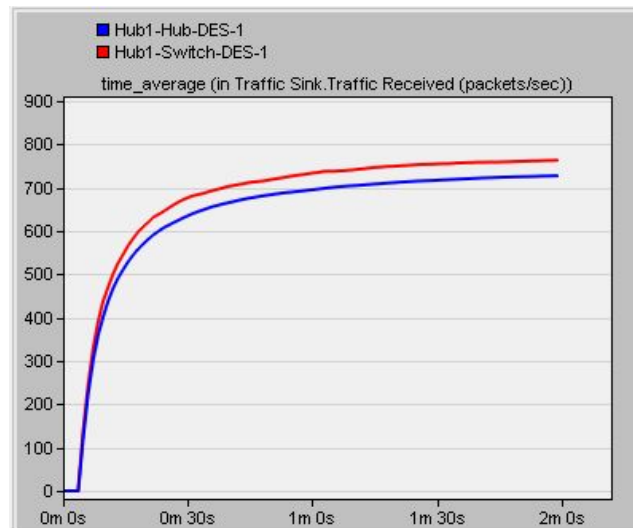
**2. Expliqueu si hi ha alguna diferència entre el escenari Hub i el escenari Switch.**

Ethernet → Delay(sec)



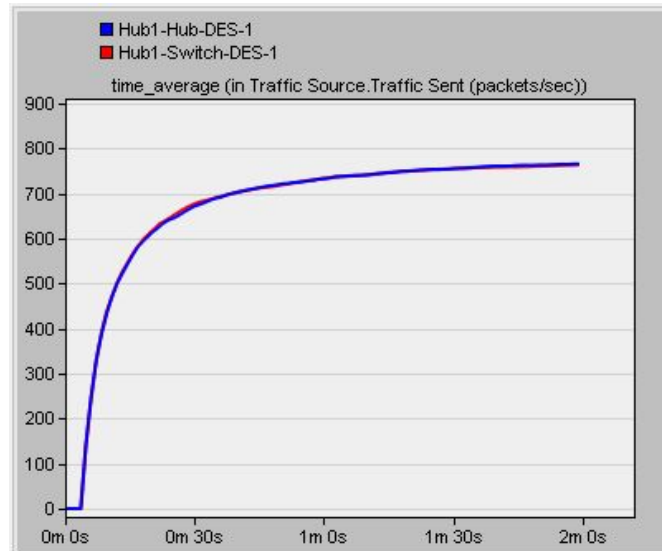
En l'escenari Switch no es produeix cap mena de delay degut a que no es possible que hi hagi col·lisions, en canvi en l'altre com ja hem esmentat si que es produeixen.

Traffic Sink → Received (packets/sec)



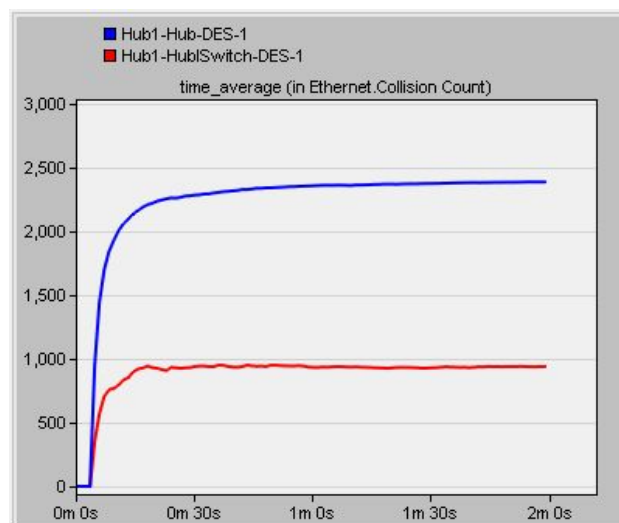
Degut a que el delay es menor en l'escenari Switch, aquest rep més paquets per segon, tot en conseqüència de que no es poden produir col·lisions, a diferència de l'altre escenari.

### Traffic Source → Traffic Sent (packets/sec)



La mitjana del temps que es tarda en enviar un paquet és la mateixa perquè un cop el Hub o el Switch han rebut el missatge, només han d'enviar-lo al destinatari, i ja no es podran produir col·lisions.

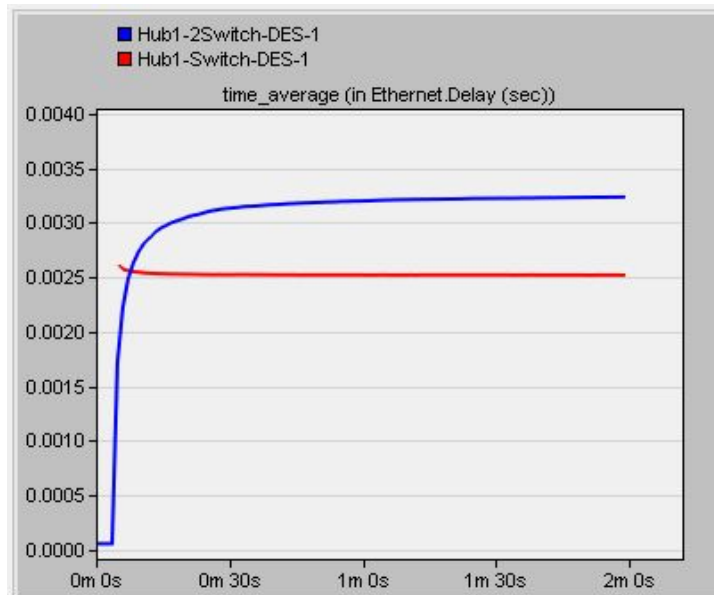
### **Observació:**



En el moment de realitzar les gràfiques de la primera i la segona pregunta, ens vam adonar de que eren pràcticament iguals, cosa que es deguda a que els paràmetres que teniem en compte són molt similars. La principal diferència es troba en la gràfica de dalt, que representa les col·lisions que es donen amb la informació enviada per un node. Com veiem, la línia blava que pertany a l'escenari Hub, té un nombre de col·lisions major que la dels altres dos escenaris. La vermella (la de l'escenari HubSwitch), té menys de la meitat de l'anterior, i en l'escenari Switch no hi ha col·lisions per tant, línia tampoc.

**3. Expliqueu si hi ha alguna diferència entre el escenari Switch i el escenari 2Switch.**

Ethernet → Delay(sec)

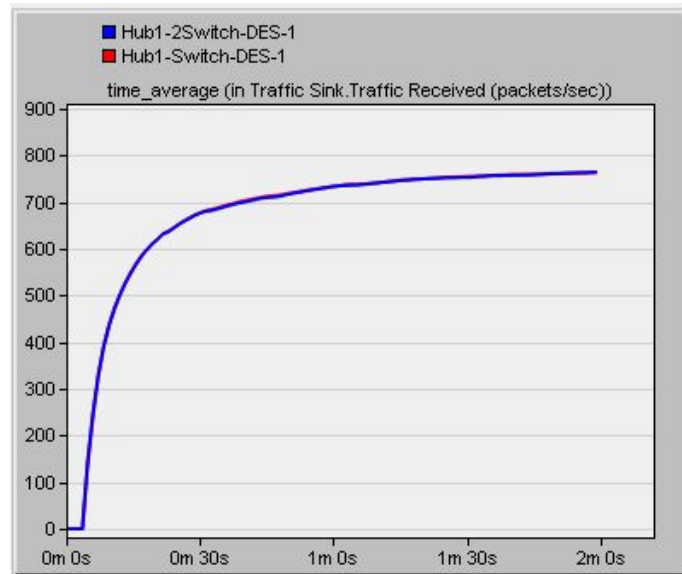


La línia vermella pertany a l'escenari Switch, i la línia blava a l'escenari 2Switch. Aquesta gràfica està explicada per diversos fenòmens:

En primer lloc, quan un Switch entra a una xarxa, desconex el nombre de nodes al que està connectat i les seves dades. Per això l'inici de les línies de la gràfica són diferents. Perquè en l'escenari Switch, ha de conèixer 16 nodes i, en canvi, en l'escenari 2Switch només 8 (perquè com tenim dos Switch, els nodes estan repartits).

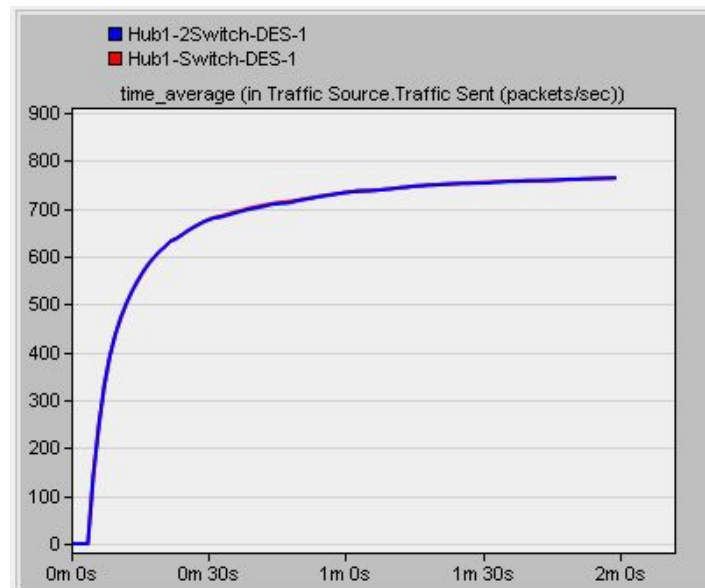
Posteriorment, veiem que el delay és major en l'escenari 2Switch. L'explicació del perquè és major, és el fet de que quan només tenim un switch (escenari Switch), els missatges només fan el camí: emissor-switch-destinatari. En canvi, si tenim dos Switch (escenari 2Switch), segons quin node sigui el destinatari, el missatge fa el procés: emissor-switch-switch-destinatari. Per tant, el fet de fer un pas més, fa que el delay augmenti.

Traffic Sink → Received (packets/sec)



Degut a que els dos escenaris utilitzen un Switch, no es produeixen col·lisions, per tant, els paquets rebuts per segon son els mateixos.

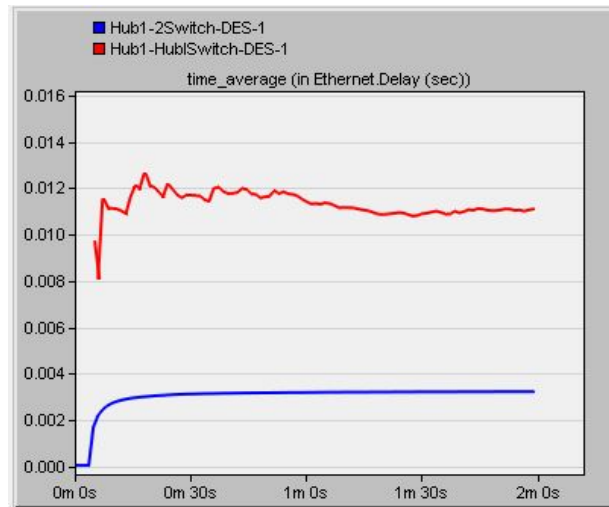
Traffic Source → Traffic Sent (packets/sec)



A causa de que les col·lisions no afecten l'enviament de paquets, les gràfiques dels 2 escenaris en aquest cas, són iguals.

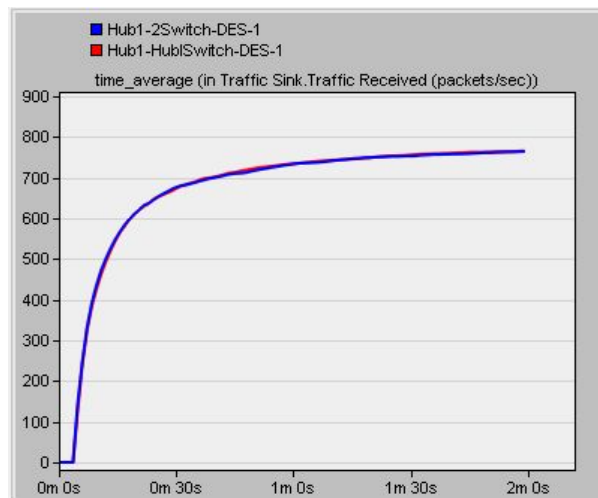
**4. Expliqueu si hi ha alguna diferència entre el escenari HublSwitch i el escenari 2Switch.**

Ethernet → Delay(sec)



Com s'observa a la gràfica, la línia que representa l'escenari HublSwitch, és força major a la que representa l'escenari 2Switch, com ja hem mencionat abans, això és degut a les col·lisions que es produeixen entre els missatges que envien els ordinador a l'hora de connectar-se a la xarxa.

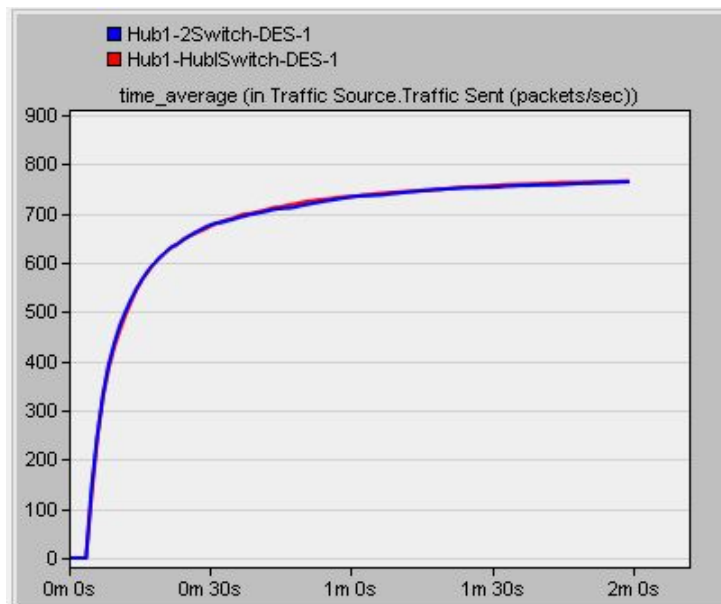
Traffic Sink → Received (packets/sec)



El traffic sink dels dos casos es pràcticament idèntic. Això és degut a que compara els escenaris, i es troba que en ambdós casos el Switch es l'element que té més pes.

En la comparació dels Switch, per tant, el temps que triguen els dos en rebre un paquet és igual, com passa també amb les col·lisions, i així queda plasmat a la gràfica.

Traffic Source → Traffic Sent (packets/sec)



Les col·lisions no afecten a l'enviament dels paquets, per tant, no es veu reflexat cap diferència entre escenaris a la gràfica.