

Fatima zahra El yamlouli
Michel Tching

Projet Réseaux Sans Fil

-

Analyse de performance du protocole TSCH et de l'ordonnancement ORCHESTRA

Introduction:

Nous avons effectué une mesure de performance du protocole TSCH. Nous avons choisi une seule topologie et deux métriques. Nous avons utilisé les configurations par défaut de TSCH et orchestra de contiki-ng.

Topologie:

Notre topologie est 2 carrés superposés de 3x3 sans le nœud central. Il y donc 16 nœuds. Parmi les 16 nœuds, il y a 1 coordinateur de PAN dans un “coin de carré” et 15 senders. Un sender envoie à intervalle régulier un message UDP de 64 octets au coordinateur de PAN. Le coordinateur de PAN n’envoie aucun message.

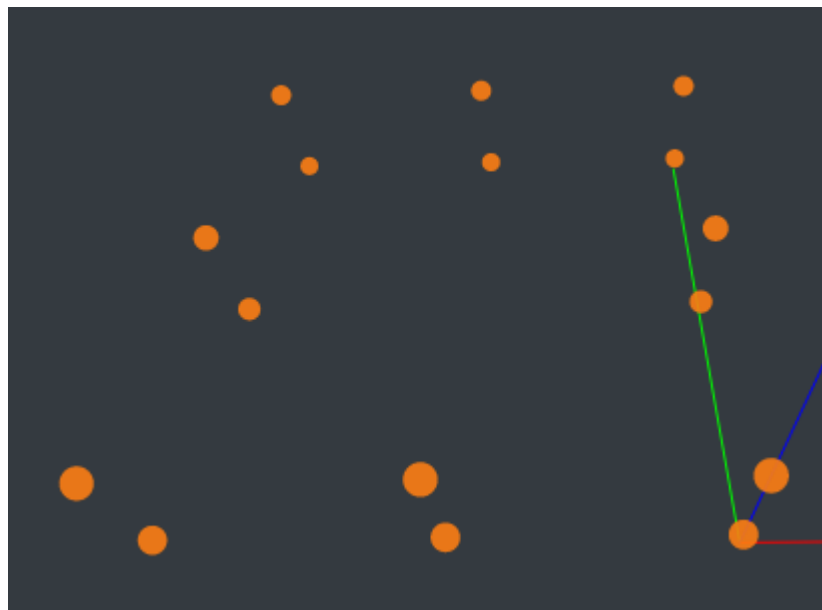


Schéma de la topologie

Cette topologie peut correspondre à un scénario où les objets connectés se trouvent sur les murs d’une salle. Ces objets envoient des mesures au coordinateur de PAN qui les stocke dans une base de données par exemple.

Métriques choisies:

Délai : Le temps que met le message pour être délivré.

PDR : Le ratio des messages qui ont bien été reçus.

Nous avons effectué les mesures à différents intervalles d'envoi : 1s, 3s, 5s, 7s, 10s.

Méthode de mesure:

Chaque message envoyé est de la forme "<adresse-mac>-<compteur>". Ce format permet de le rendre unique. À l'envoi et à la réception de message, les nœuds affichent sur la sortie standard un message de la forme "<type> <adresse-mac>-<compteur>" où type est RX ou TX. Cela permet d'identifier l'émission et la réception pour chaque message.

Le serial_aggregator permet de réunir toutes les sorties standards sur un seul terminal. Il ajoute aussi au message de sortie un temps et le nœud qui affiche le message.

Nous pouvons alors mesurer le délai par une simple différence entre l'heure de réception et l'heure d'émission. Et le PDR peut être mesuré car nous distinguons les émissions des réceptions.

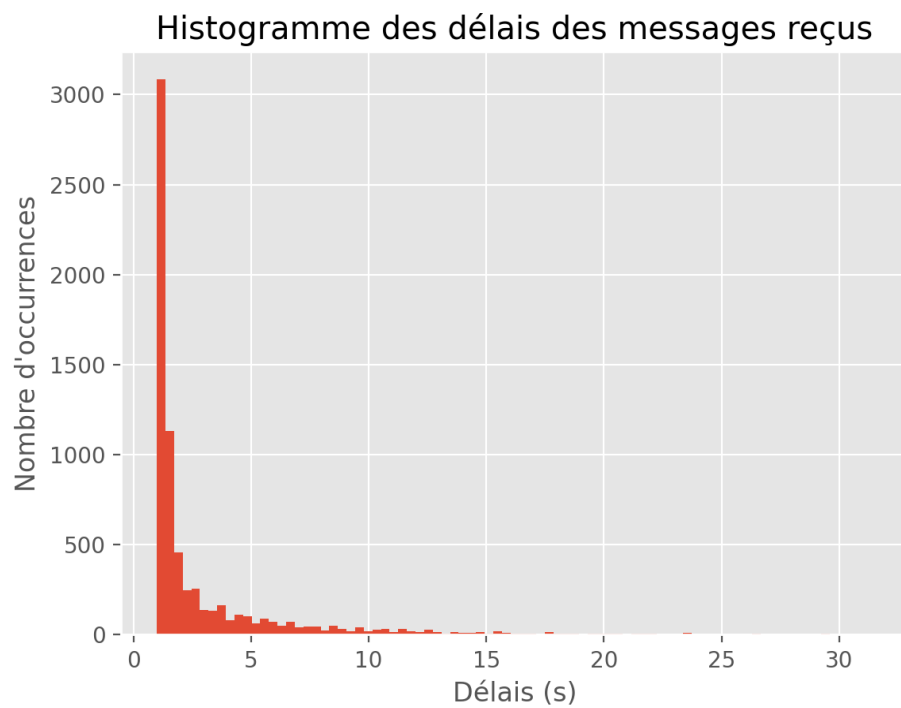
La charge du trafic est calculée en utilisant la fréquence

Les expériences durent 20 minutes.

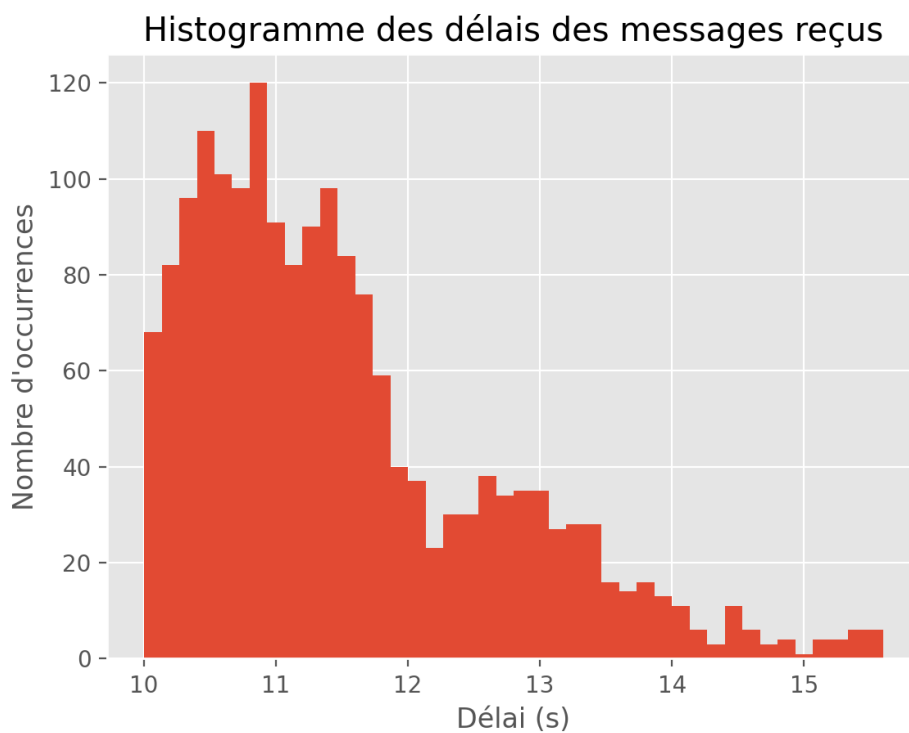
Histogramme des délais:

Les histogrammes montrent que les variations du délai sont proportionnelles à la charge du trafic.

Dans le cas de la forte charge de trafic, la majorité des messages reçus sont ceux avec un délai court. Cela est expliqué par le fait que les autres messages sont jetés car les buffers sont remplis.



charge du trafic :
7680 bps



charge du trafic:
768 bps

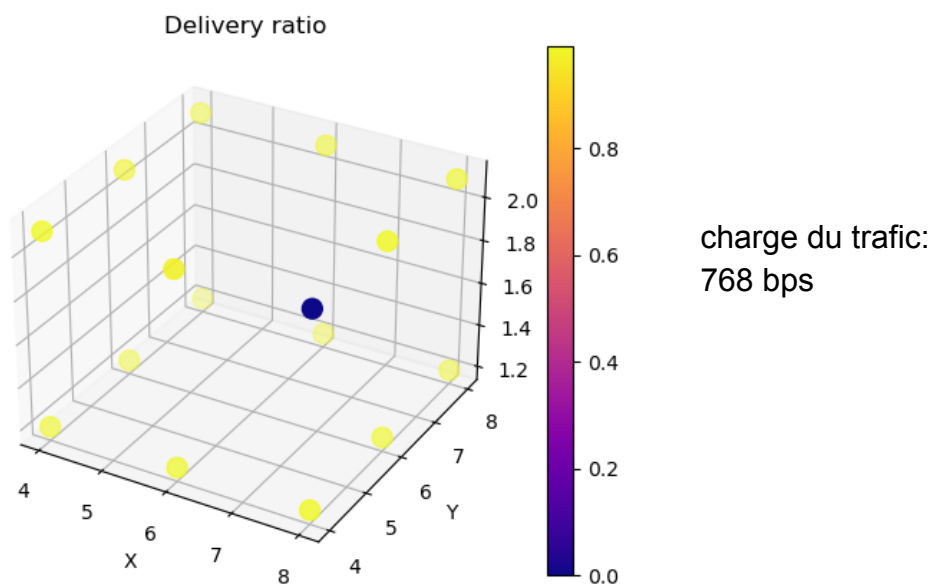
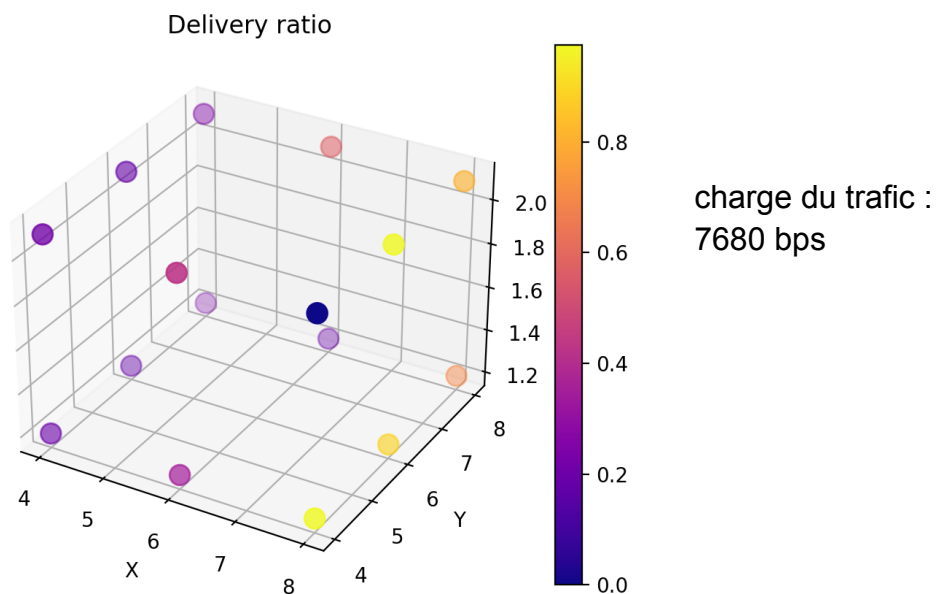
PDR:

Le nœud bleu foncé à 0 correspond au coordinateur de PAN.

Les PDR de tous les nœuds du trafic faible est presque égal à 1, tandis que ceux du trafic fort sont variables.

L'augmentation de la charge du trafic diminue le PDR des nœuds. Le réseau ne supporte donc pas complètement cette charge. Cela confirme que les messages ont bien été jetés.

Nous observons que dans l'expérience avec le trafic fort, les nœuds avec une plus grande coordonnée x ont de meilleur PDR. Nous ne savons pas comment l'expliquer.

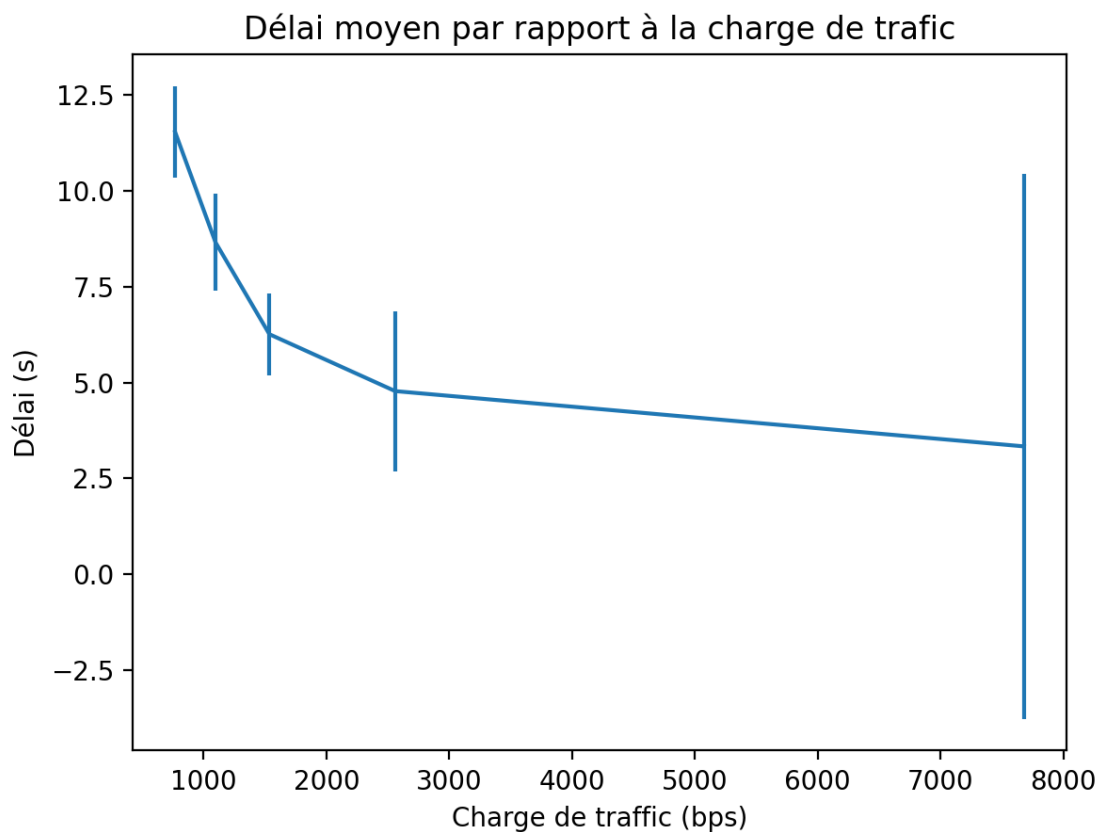


Délais moyen par rapport à la charge du trafic:

Le graphe montre que le délai moyen diminue quand le trafic augmente. Mais que l'écart type augmente aussi et donc la variation du délai augmente.

Ce résultat concorde avec les résultats précédents. En effet, l'écart-type augmente car les variations sont plus importantes.

Le délai moyen diminue car on ne prend en compte que les messages reçus. En effet, ces messages jetés augmenteraient le délai moyen si le buffer était plus grand.



Fichiers utilisés:

programmes des noeuds : coodinator.c, simple-sender.c

automatisation des expériences : square.sh

visualisation: occ.py, pdr.py, throughput.py

images : dossiers images

donnée brute : dossier raw_data

Difficultés rencontrées:

Nous avons des difficultés à prendre en main contiki-ng. La recherche dans la documentation et les exemples a pris énormément de temps.

Nous avons dû changer de stratégie d'envoi des messages. Le code de sender.c devait choisir des destinations aléatoires mais seul le coordinateur de PAN était choisi. Cela est peut être dû au fait que nous utilisons RPL_LITE qui ne sauvegarde pas les autres nœuds. RPL_CLASSIC ne fonctionnait pas, les nœuds ne rejoignaient pas le réseau.