

Manuel d'utilisation de l'outil

MULANE

FRA - ANG

Table des matières

Utilisation de l’outil MULANE	3
1. A propos de l’outil MULANE	3
2. Configuration requise	4
Exploration de l’outil	5
1. Fonctionnement de l’outil	5
2. Page de démarrage	5
3. Lancement d’une simulation	6
3.1 Choix des paramètres fixes	6
3.2 Choix de la méthode	8
3.2.1 Les méthodes présentes dans Mulane	8
3.2.2 Choix de la méthode	11
3.2.3 Choix des paramètres d’une méthode	12
4. Résultats de la simulation	13
4.1 Résultats sous forme graphique	13
4.2 Présentation des résultats	14
5. Conclusion	18

Utilisation de l'outil MULANE

1. A propos de l'outil MULANE

MULANE, qui veut dire Multi-methods based spreading factor for LoRa Networks est un outil qui permet de simuler avec plusieurs méthodes d'amélioration des performances des réseaux LoRaWAN.

LoRaWAN est un protocole de télécommunication utilisant LoRa qui est une technologie qui permet la communication entre divers objets par le biais d'une connexion internet et ainsi réaliser un réseau d'objets.

Mais un réseau LoRaWAN peut rencontrer quatre types de collisions :

Frequency collision :

Une collision de fréquences arrive quand deux paquets ont une fréquence trop proche, on peut résumer cela en trois cas :

- Si la différence de fréquences des deux paquets est inférieure ou égale à 120KHz et la bande passante des deux paquets est égale à 500.
- Si la différence de fréquences des deux paquets est inférieure ou égale à 60KHz et la bande passante des deux paquets est égale à 250
- Si la différence de fréquences des deux paquets est inférieure ou égale à 30KHz

SF collision

Si les valeurs des SFs des deux paquets sont égaux alors une collision de type SF collision est détectée.

Energy collision

Une collision de tel type détectée, si l'un des cas suivants est réalisé :

- La valeur absolue de la différence entre le rssi (Received Signal Strength Indication) qui est une mesure du niveau de puissance en réception d'un signal issu d'une antenne des deux paquets est inférieure à un seuil de 6 dB (La puissance seuil fonctionnelle), dans ce cas les deux paquets sont trop proches l'un de l'autre, les deux entrent en collision, renvoyer les deux paquets comme victimes, Sinon ;
- Si, la différence entre le rssi des deux paquets est strictement inférieure à un seuil de 6 dB ($p1.rssi - p2.rssi < (\text{La puissance seuil fonctionnelle})$), dans ce cas $p2$ surpuissant $p1$, renvoyer $p1$ comme victime.
- Sinon ; $p2$ était le paquet le plus faible, renvoyer $p2$ comme victime

Timing collision

En supposant que $p1$ est le paquet fraîchement arrivé et que c'est la dernière vérification, nous avons déjà déterminé que $p1$ est un paquet faible, donc la seule façon de gagner est d'être suffisamment en retard (seuls les $n-5$ premiers symboles de préambule se chevauchent).

En supposant que 8 symboles de préambule le rendent comme une victime.

Mulane est là pour présenter plusieurs méthodes qui permettent d'augmenter les performances des réseaux LoRaWAN et apprendre à utiliser cet outil n'a rien de compliqué, mais cela suppose de bien comprendre les méthodes intégrer. Nous vous conseillons d'étudier ce manuel en détail avant d'utiliser l'outil.

Ce manuel de l'utilisateur offre une vue d'ensemble des caractéristiques de l'application et donne des instructions pas à pas pour la réalisation de différentes tâches.

2. Configuration requise

- Avant d'utiliser MULANE, veuillez-vous assurer que Python 3 est installer sur votre machine avec comme variable d'environnement "python".
- Pour installer les bibliothèques nécessaires lancer le fichier "requirements.txt" avec la commande "pip install -r requirements.txt" avant le premier usage.

Pour lancer l'outil, ouvrir le terminal dans le dossier qui contient "mulane.py" avec la commande "python mulane.py".

Exploration de l'outil

1. Fonctionnement de l'outil

L'outil fonctionne en 4 étapes :

- **Choix des paramètres fixes** L'utilisateur choisit les paramètres fixes qui ne change pas pour toutes les méthodes.
- **Choix de la méthode** L'utilisateur choisit la méthode avec laquelle il veut effectuer des simulations.
- **Choix des paramètres** Chaque méthode a des paramètres spécifiques que l'utilisateur peut définir.
- **Paramètres graphiques** Avant de lancer la simulation, on peut choisir si on veut représenter les résultats sous forme de graphiques ou texte uniquement.

2. Page de démarrage

La page de démarrage constitue une plateforme pratique qui apporte des informations et permet d'atteindre les différentes parties de l'outil.

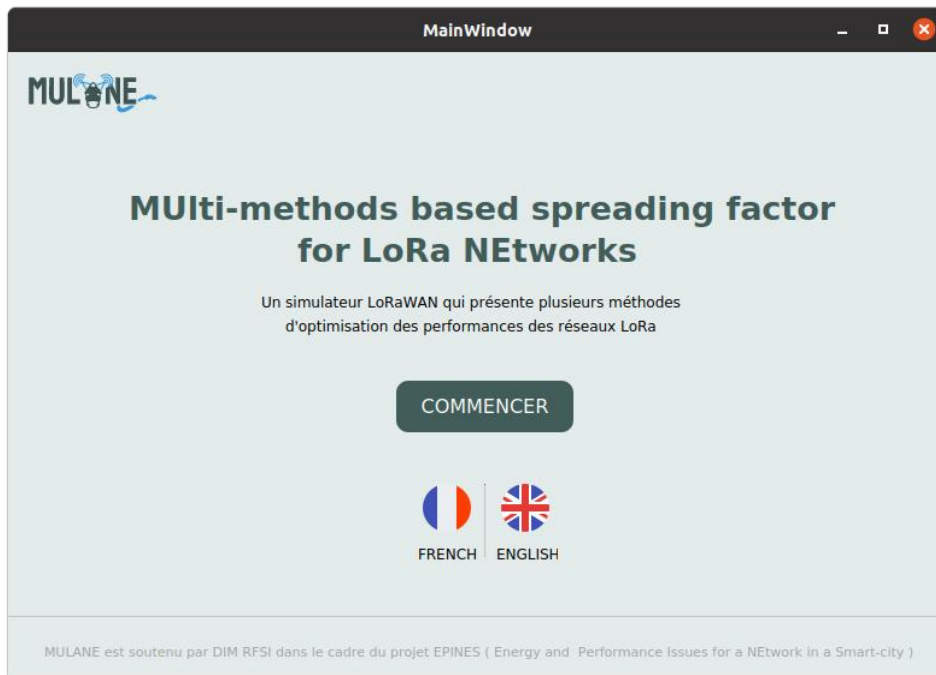


Figure (1) : fenêtre d'accueil outil Mulane

- **Pour choisir la langue**, cliquez sur le drapeau de la langue souhaité
- **Pour démarrer une nouvelle simulation**, cliquez sur Begin en Anglais sinon sur Commencer en Français.

3. Lancement d'une simulation

3.1 Choix des paramètres fixes

Les différentes méthodes ont des paramètres en commun qui ne change pas d'une méthode à l'autre et la page des choix des paramètres fixes permet de les initialiser.



Figure (2) : fenêtre choix des paramètres fixes.

Les paramètres fixes sont :

- **Nombre de nœuds**

Désigne le nombre de nœuds dans le réseau, chaque nœud sera placé dans le réseau en fonction des paramètres **Position des nœuds** et **Taille maximale**.

- **Position des nœuds**

Sa valeur peut aller entre 0 et 50, ce nombre représente le nombre de cercle autour de la station de base où les nœuds seront placés donc à 0 le placement des nœuds est totalement aléatoire et plus le nombre est plus grand le placement est de moins en moins aléatoire.

- **Taille maximale**

La valeur de ce paramètre est entre 400 (m) jusqu'à la distance maximale qui est de 8921.359 (m) et fixe tout simplement la distance max où les nœuds pourront être placés.

En plus des autres paramètres qui sont Taille des paquets, Temps moyens d'envoi de paquet et durée de simulation.

- Les “i” à côté des champs de texte servent à apporter des explications par rapport au paramètre
- **Pour revenir en arrière**, appuyez sur Back en Précédent.
- **Pour aller vers la prochaine étape**, appuyez sur Suivant.

3.2 Choix de la méthode

3.2.1 Les méthodes présentes dans Mulane

Mulane présente plusieurs méthodes d'amélioration des performances, dans cette version il comporte 7 méthodes.

- **Static Random**

Cette méthode se rapproche beaucoup du simulateur de base LoRaSIM avec une seule modification c'est l'initialisation du SF qui se fait d'une manière aléatoire.

- **Dynamic Random**

La méthode fonctionne avec une contrainte sur le taux de collisions, son fonctionnement est simple, chaque nœud de la station de base transmet des paquets, si un paquet tombe en collision on envoie le paquet en utilisant une autre SF choisie au hasard parmi les SF de la station.

- **Dynamic P-Random**

Son fonctionnement ressemble à celui de Dynamic Random mais en utilisant une valeur supplémentaire qui est le P dont la valeur est choisie par l'utilisateur, la valeur qu'on a choisie est 0.4. Comme d'habitude, les nœuds de la station de base transmettent des paquets, si un de nos paquets rencontre une collision, alors on tire une valeur au hasard qui varie entre 0 et 1, si le nombre tiré est supérieur ou égal à P, on change de SF sinon on ne fait rien et on force la transmission de notre paquet avec le SF choisi initialement.

- **STEPS Full et STEPS Individual**

Dans cette méthode on a introduit une nouvelle idée qui est le tableau des scores, pour faire simple, on a un tableau qui représente les probabilités de choisir un SF et la somme des valeurs du tableau sera 1 et lors d'une collision

le choix du SF se fera donc de manière aléatoire selon les probabilités du tableau.

Chaque collision rencontrée en utilisant un SF, la probabilité de ce dernier sera diminuée dans notre tableau, et dans le cas contraire elle sera augmentée.

La somme du tableau sera toujours égale à 1 car on effectue une normalisation après chaque modification des valeurs.

Lors d'une collision le SF sera donc modifié pour récupérer le meilleur SF, et s'il y a plusieurs collisions d'affilée sur le même SF, réduire encore plus son poids pour éviter de continuer de l'utiliser.

Notre tableau de poids permet de choisir le meilleur paramètre de SF lors de la transmission, il est donc envisageable de récupérer ces données avec pour objectif de les transmettre aux nouveaux nœuds lors de leur création dans les futures simulations.

L'objectif est d'utiliser ces données lors de la mise en place d'une nouvelle simulation c'est pourquoi lors de la création de nœuds nous récupérerons les données du nœud le plus proche du nouveau nœud qui sont stockées dans une base de données.

STEPS Full et STEPS Individual sont les mêmes simulateurs dans leurs fonctionnements, la différence est que STEPS Individual est un simulateur dans des conditions plus réel que STEPS Full car il a moins d'informations sur les paquets qu'il a envoyé (il ne sait pas si un paquet a été reçu que s'il y a un ack).

STEPS E-Greedy et Boltzmann

Calcul de SF minimale : Le SF minimal représente le SF $\in \{7,8,9,10,11,12\}$ pour lequel le récepteur pourra recevoir la transmission ce SF dépend de la distance entre le nœud et la station de base c'est pour cela qu'il faudra nous intéresser à la puissance de réception, qui dépend de la puissance de transmission, et à la sensibilité du récepteur, qui dépend du SF et de la bande passante car en effet la transmission ne pourra être reçue avec succès que si la puissance de réception est supérieure à la sensibilité.

Dans un premier temps il faut observer la puissance de réception P_{rx} qui nous est donné avec l'équation : $P_{rx} = P_{tx} + GL - L_{pl}$

Où P_{tx} est la puissance de transmission qui est paramétrable au niveau de chaque transmetteur et est théoriquement ajustable de -4dBm à 20 dBm par pas de 1dB mais est limité par les implémentations hardware et les régulations de fréquence. GL est la somme des gains et des pertes de transmission et de réception et L_{pl} est le calcul de la perte de chemin obtenu suivant : $L_{pl} = L_{pld0} + 10n\log_{10}(d/d_0)$.

Où L_{pd0} et n sont respectivement la perte à la distance de référence d_0 et le coefficient de perte de chemin et sont obtenu de manière expérimental, d est la distance entre le nœud et la station de base.

Maintenant que nous avons vu comment est obtenu la puissance de réception il nous faut vérifier que celle-ci soit bien supérieur à la sensibilité du récepteur, la sensibilité dépend du SF choisi ainsi que de la bande passante. En utilisant la calculatrice LoRa SX1272 disponible à ce lien nous obtenons le tableau de sensibilité suivant (en dBm) :

		Bandwidth		
		125	250	500
SF	7	-123	-120	-117
	8	-126	-123	-120
	9	-129	-126	-123
	10	-132	-129	-126
	11	-134.5	-131.5	-128.5
	12	-137	-134	-131

Sensibilité calculé du receveur pour différent SF et BW

Le SF minimal est donc le plus petit SF pour lequel la puissance de réception est supérieur à la sensibilité du receveur, on a donc pour chaque nœud un ensemble de SF disponible S tel que $S = [S_{fmin}, 12]$

Estimation des récompenses : Chaque nœud utilise un tableau contenant des estimations de récompenses, r_i pour chacun des 6 SFs disponible, le tableau est initialisé soit à partir des expériences de la base de données soit en fonction de la distance du nœud à la passerelle si la base de données est vide. A chaque transmission le nœud recevra une récompense R égale à 1 en cas d'acquiescement et à 0 dans tous les autres cas. Après la transmission l'estimation de récompense E_{ri} pour i choisi dans la transmission sera mise à jour d'après la formule : $E_{ri} = E_{ri} + \alpha (R - E_{ri})$

Où α est un facteur d'apprentissage et $\alpha \in (0, 1]$ choisit par l'utilisateur.

E-Greedy : Lors de la sélection d'un SF pour la transmission, un $SF \in S$ sera choisi de manière aléatoire avec une probabilité ϵ , sinon le SF ayant la meilleur E_r sera choisi. On a donc P_i la probabilité de choisir le SF i tel que :

$$P_i = \begin{cases} 1 - \varepsilon + \frac{\varepsilon}{K} & \text{si } i = \operatorname{argmax} E r_i \\ \frac{\varepsilon}{K} & \text{sinon} \end{cases}$$

Où $\varepsilon \in (0, 1]$ est choisi par l'utilisateur et K est le cardinal de S .

Boltzmann exploration : Lors de la sélection d'un SF pour la transmission, le SF sera choisi de manière aléatoire selon une probabilité P_i dépendant de l'estimation $E r_i$ de chaque SF ainsi que d'un paramètre $\tau \in \mathbb{R}^+$ tel que :

$$P_i = \frac{e^{\frac{E r_i}{\tau}}}{\sum_{k=SF_{min}-\tau}^{\tau} e^{\frac{E r_k}{\tau}}}$$

Où τ est choisi par l'utilisateur, plus τ s'approche de l'infini plus l'algorithme se comporte de manière aléatoire et plus τ est proche de 0 plus l'algorithme se comporte comme un algorithme greedy.

Pour tout $S F_i < S F_{min}$ nous aurons $P_i = 0$.

3.2.2 Choix de la méthode



Figure (3) : fenêtre choix de méthode

- La fenêtre permet à l'utilisateur de choisir la méthode grâce à un menu déroulant (le menu déroulant a été choisis pour la représentation des différentes méthodes afin de permettre de rajouter de nouvelles méthodes dans le futur)
- Lorsqu'une méthode est sélectionnée, une description de la méthode apparait juste en dessous pour expliquer à l'utilisateur les spécificités de la méthode
- **Pour revenir en arrière**, appuyez sur Back en Précédent.
- **Pour aller vers la prochaine étape**, appuyez sur Suivant.

3.2.3 Choix des paramètres d'une méthode

Chaque méthode a des paramètres spécifiques alors tout dépend de la méthode choisie, les paramètres ne sont pas les mêmes.



Figure (4) : fenêtre choix des paramètres d'une méthode

Prenons la méthode STEPS_full comme exemple car elle comporte la plupart des paramètres saisissables qui sont :

- **Modèle de collision**

Ou mode de détection des collisions, on a deux modes de détection, collision complète et collision simple

On sait qu'il existe quatre types de collisions (Frequency collision, SF collision, timing collision et Energy collision)

Le mode détection complète détectera tous ces types de collision alors que le mode collision simple ne détectera que les collisions de type (Frequency collision et SF collision).

- **Nombre de lancement**

Le nombre de fois que le simulateur sera lancé et les résultats seront la moyenne par rapport à la valeur de ce paramètre.

- **Taille de la base de données (BD) souhaitée**

Représente la taille de BD que le simulateur démarre avec, si par exemple l'utilisateur choisi une valeur de 100, le simulateur va préparer une BD de 100 entrées et puis commencer la simulation.

- **Mise à jour de la BD**

Durant la simulation l'utilisateur peut choisir de permettre la maj de la BD, c'est-à-dire permettre le partage d'expérience entre les nœuds ou bien de ne pas permettre la maj de BD afin de simuler sur un nombre d'entrées de BD fixe.

- L'utilisateur remplit les champs
- Là aussi si tous les champs ne sont pas remplis, l'utilisateur n'a pas le droit de passer à l'étape suivante.

4. Résultats de la simulation

4.1 Résultats sous forme graphique

L'utilisateur peut choisir de générer les résultats sous forme graphique grâce à la page des paramètres graphique qui permet de cocher les résultats souhaités avoir sous forme graphique.

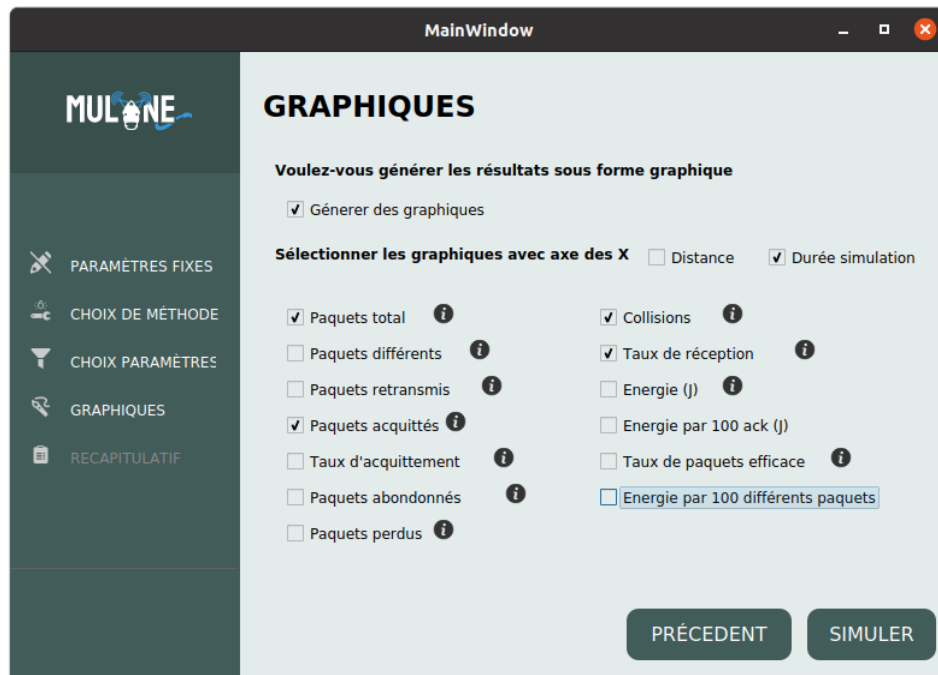


Figure (5) : fenêtre paramètres graphique

- L'utilisateur a le choix de soit générer des graphiques avec les résultats ou non
- S'il choisit d'en générer il pourra choisir lesquelles générer
- Il peut aussi choisir de générer les graphiques soit en fonction de la distance ou bien de la durée de simulation et même les deux au même temps
- Si l'axe des X n'est pas choisis, les deux seront générer

4.2 Présentation des résultats

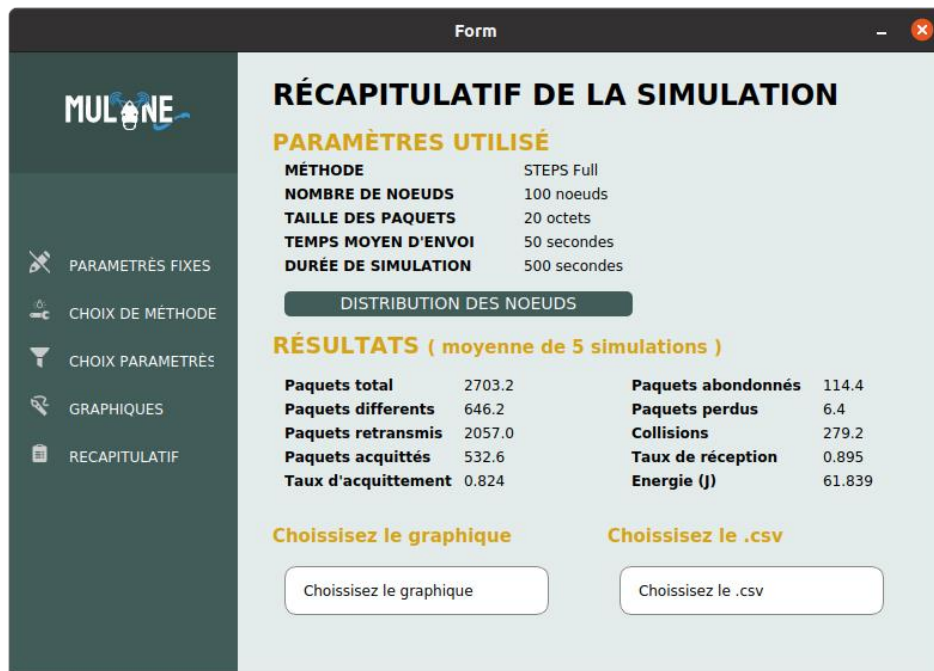


Figure (6) : fenêtre résultats de la simulation

La fenêtre affiche dans une première partie un rappel des paramètres utilisés ainsi que la distribution des nœuds par rapport à la station de base et le SF utilisé par chaque nœud.

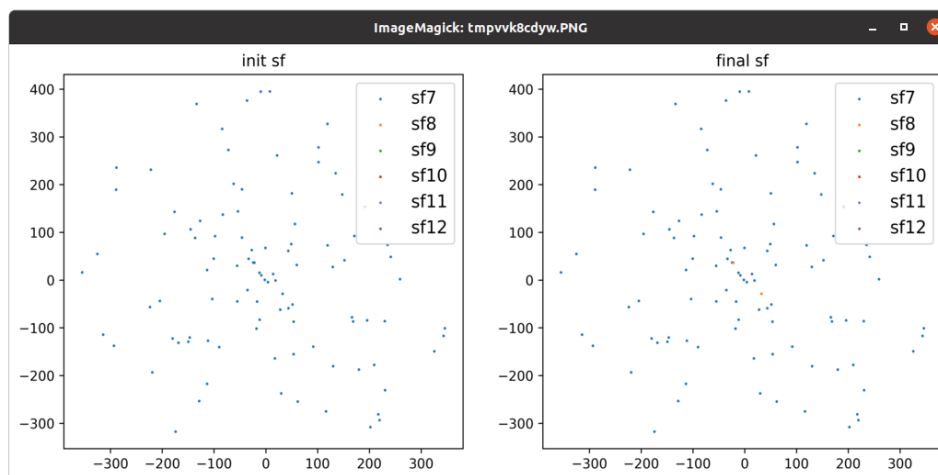


Figure (7) : fenêtre répartition des nœuds et leurs SF.

Dans la deuxième partie les résultats de la simulation à savoir :

- **Paquets total** : Le nombre de transmissions total effectuée par les nœuds du réseau on peut dire que (paquets total = paquets différents + paquets retransmis).
- **Paquets différents** : Le nombre de paquets unique envoyé par les nœuds du réseau.
- **Paquets retransmis** : le nombre de retransmission de paquets effectuée par les nœuds du réseau, un paquet est retransmis si sa première transmission n'était pas concluante.
- **Paquets acquittés** : Le nombre de paquets acquittés par la station de base.
- **Taux d'acquiescement** : Représente le nombre de paquets acquittés / nombre de paquets différents.
- **Paquets abandonnés** : nombre de paquets abandonnés par les nœuds, un paquet est abandonné s'il a été retransmis plus de 8 fois.
- **Paquets perdus** : Nombre de paquets perdus dans le réseau pendant la transmission.
- **Collisions** : Nombre de collisions détecté dans le réseau durant la transmission.
- **Taux de réception** : Représente le nombre de paquets reçus / nombre de paquets total.
- **Energie** : L'énergie consommé par les nœuds pour effectuer les transmissions nécessaires.

Dans la troisième partie l'utilisateur peut choisir le graphique qu'il veut afficher en plus des fichiers .CSV afin de lui permettre de vérifier les valeurs.

L'utilisateur pourra donc visualiser les différents graphiques qu'il a cocher depuis le menu déroulant, par exemple dans notre simulation on a choisi de générer les graphiques de : paquets total, paquets acquittés, collisions et taux de réception.

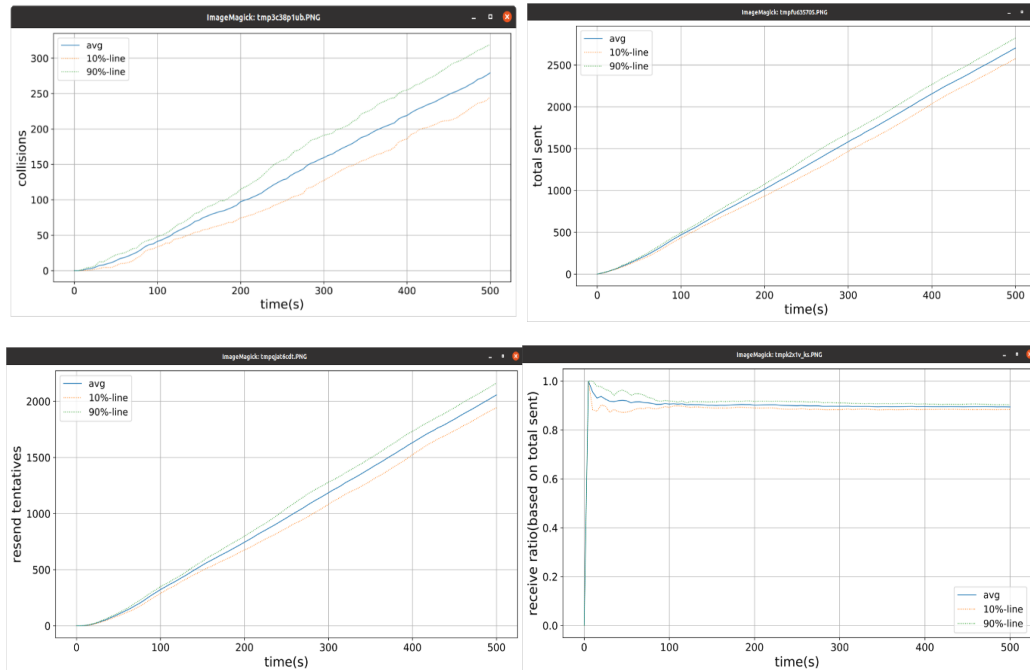


Figure (7) : représentation graphique des résultats

Et de même pour les fichiers .CSV, il faut savoir que pour chaque graphique générer, son fichier .CSV aussi il est généré et l'utilisateur pourra visualiser les données avec lesquelles les graphiques ont été dessinés, par la suite il pourra soit vérifier les données, dessiner un autre type de graphique à l'aide de ses données.

	A	B
1	sim time	collisions
2	0	0
3	5000	2
4	10000	2
5	15000	2
6	20000	2
7	25000	4
8	30000	4
9	35000	4
10	40000	4
11	45000	4
12	50000	5
13	55000	5
14	60000	5
15	65000	6
16	70000	6
17	75000	8
18	80000	9
19	85000	13
20	90000	13
21	95000	13
22	100000	13
23	105000	13

	A	B	C	D
1	x	y	distance	collisions
2	-19.54017628	-23.45116542	30.52500039	1
3	214.4471867	-14.00328005	214.9039034	0
4	12.78123521	155.0745936	155.6004165	0
5	81.95404787	332.1726193	342.1331831	0
6	9.979289283	8.880503129	13.35850105	0
7	-32.76900868	-169.1656209	172.3102296	0
8	-222.9247223	244.945408	331.2003694	4
9	110.1513331	-291.5468964	311.6615295	2
10	136.6529455	-114.778751	178.4606096	2
11	-2.948895646	133.7589595	133.7914617	1
12	339.1876743	-51.87423013	343.1314823	0
13	-121.2951936	-25.02496377	123.8497993	3
14	116.7039284	-144.7745413	185.9555719	0
15	25.53573676	8.565486931	26.93401972	0
16	-50.14135651	41.87195857	65.3254663	2
17	257.3976922	194.6862998	322.7325941	0
18	62.95818954	-299.1027204	305.6569498	1
19	-52.42164265	-166.7133331	174.7608768	0
20	111.9083397	342.2204934	360.0532497	1
21	332.3745308	-7.572232048	332.4607758	3
22	2.030091914	-264.2418852	264.2496834	1
23	303.9815645	-172.5273252	349.5289251	2

Figure (8) : exemple de fichier CSV des résultats

5. Conclusion

L'outil Mulane permet de simuler avec différentes méthodes créées pour optimiser les performances des réseaux LoRaWAN.

Un deuxième outil "Mulane & Comparaison" a été créé pour permettre de comparer entre les méthodes de base (LoRaSIM, LoRaFREE et LoRaMAB) et toutes les nouvelles méthodes de Mulane.