

# Évaluation de Performance TP2

Chen LU, Yizhi YANG

Octobre 2024

## 1 Aperçu de l'Expérience

Nous avons mené quatre tâches principales pour évaluer les performances d'E/S dans différentes conditions :

Tâche 1 : Variation du pourcentage d'écriture pour des schémas d'accès séquentiels et aléatoires.

Tâche 2 : Variation de la taille des blocs de 1 Ko à 1 Mo.

Tâche 3 : Utilisation d'une distribution mixte de tailles de blocs pour les opérations d'écriture.

Tâche 4 : Variation du nombre de travaux parallèles.

Chaque tâche impliquait l'exécution de multiples tests Fio avec des configurations de paramètres spécifiques, en répétant chaque test plusieurs fois pour recueillir suffisamment de données pour une analyse statistique.

## 2 Configuration Générale de l'Expérience

La préparation de Fio comprend la création d'un fichier de test de 1 Go nommé testfile, utilisé comme cible d'E/S pour tous les tests, la configuration des paramètres par défaut spécifiés dans le fichier default.fio, incluant l'engine d'E/S, l'I/O direct, la durée d'exécution, la taille de bloc et la taille de fichier, ainsi que la mise en mode maximale de la performance des CPUs pour stabiliser les mesures.

## 3 Détails des Tâches

### 3.1 Tâche 1 : Variation du Pourcentage d'Écriture avec Différents Schémas d'Accès

L'objectif de cette tâche est d'analyser l'impact de la variation du pourcentage d'écriture sur la bande passante d'E/S pour des schémas d'accès séquentiel et aléatoire. Les schémas d'accès évalués sont séquentiel (rw) et aléatoire (randrw), et les pourcentages d'écriture testés incluent 0%, 12%, 25%, 38%, 50%, 63%, 75%, 88% et 100%. Pour chaque configuration, une boucle externe itère sur les schémas d'accès, une boucle intermédiaire sur les pourcentages d'écriture et une boucle interne effectue dix répétitions. Les commandes Fio sont ajustées avec les options -rw pour le schéma d'accès et -rwmixwrite pour le pourcentage d'écriture courant, en utilisant les autres paramètres de default.fio. Des journaux de bande passante (\*bw.log) ont été générés et les données extraites pour l'analyse. En tout, 180 exécutions ont été réalisées (2 schémas d'accès  $\times$  9 pourcentages d'écriture  $\times$  10 répétitions).

### 3.2 Tâche 2 : Variation de la Taille de Bloc

L'objectif de cette tâche est d'examiner l'impact de différentes tailles de blocs sur les performances d'E/S. Les tailles de blocs testées incluent 1 Ko, 2 Ko, 4 Ko, 8 Ko, 16 Ko, 32 Ko, 64 Ko, 128 Ko, 256 Ko, 512 Ko et 1024 Ko. Pour chaque taille de bloc, le test a été répété dix fois. La commande Fio est configurée avec l'option -bs pour la taille de bloc courante, tandis que les autres paramètres proviennent de 'default.fio'.

Les journaux de bande passante ont été générés et traités de manière similaire à la Tâche 1. En tout, 110 exécutions ont été réalisées (11 tailles de blocs  $\times$  10 répétitions).

### 3.3 Tâche 3 : Requêtes d'Écriture avec Tailles de Blocs Mixtes

L'objectif de cette tâche est d'évaluer les performances lors de l'utilisation d'un mélange de différentes tailles de blocs pour les opérations d'écriture. La distribution des tailles de blocs (bssplit) est configurée comme suit : 30% de requêtes de 4 Ko, 60% de requêtes de 16 Ko et 10% de requêtes de 64 Ko. Le paramètre `-bssplit` est défini sur `4k/30 :16k/60 :64k/10`, et `-rw` est réglé sur `write`. Le test a été répété dix fois. Les journaux de bande passante et de latence ont été collectés, incluant la latence totale, la latence de complétion et la latence de soumission. En tout, 10 exécutions ont été réalisées.

### 3.4 Tâche 4 : Variation du Nombre de Travaux Parallèles

L'objectif de cette tâche est de comprendre l'impact de l'augmentation du nombre de travaux parallèles sur les performances d'E/S. Les valeurs testées pour le nombre de travaux sont 1, 2, 4, 6 et 8. Pour chaque configuration, le test a été répété dix fois. La commande `Fio` est configurée avec l'option `-numjobs`, définie sur le nombre de travaux courant. Les journaux de bande passante ont été collectés et traités de la même manière que dans les tâches précédentes. En tout, 50 exécutions ont été réalisées (5 nombres de travaux  $\times$  10 répétitions).

## 4 Analyse des Données

En premier lieu, nous utilisons la méthode ANOVA pour comparer les résultats de plusieurs tests dans des conditions identiques (`same.cond.py`). Dans la plupart des cas, la valeur `F` est faible et la valeur `p` est supérieure à 0,05, ce qui indique que les résultats sont relativement stables. Ensuite, en comparant les résultats des tests dans des conditions différentes, une valeur `F` plus élevée et une valeur `p` inférieure à 0,05 montrent que les variables de condition ont un certain impact sur les résultats des tests.

L'analyse des données a été réalisée à l'aide de scripts Python : le script `compare.py` effectue des analyses statistiques (ANOVA) pour évaluer si les différences de configurations ont un impact significatif sur les performances d'E/S; `compare.s.py` est similaire à `compare.py` mais réalise des analyses statistiques individuellement pour chaque répétition. Le script `anova_avg.py` calcule la moyenne empirique de chaque test et applique l'ANOVA uniquement sur les moyennes des tests de 1 à 10. Enfin, `plot.py` génère des graphiques permettant de visualiser les données, notamment les tendances de bande passante et les distributions de latence.

## 5 Méthodes Statistiques

L'analyse a impliqué une ANOVA (Analyse de Variance) pour déterminer s'il existait des différences statistiquement significatives de bande passante entre diverses configurations, une extraction des données où les valeurs de bande passante et de latence ont été récupérées des fichiers journaux et organisées pour l'analyse, et des mesures statistiques incluant la moyenne (bande passante ou latence moyenne), la médiane (valeur centrale permettant une estimation moins sensible aux valeurs aberrantes) et l'écart-type (mesure de la dispersion des données); une régression linéaire a également été effectuée pour identifier les tendances et les relations entre les paramètres et les mesures de performance.

## 6 Génération des Graphiques

Le script `plot.py` a généré plusieurs graphiques pour chaque tâche, enregistrés avec des noms de fichiers spécifiques.

## 6.1 Graphiques de la Tâche 1

Les graphiques de la tâche 1 présentent les résultats pour les accès séquentiel (rw) et aléatoire (randrw). Pour l'accès séquentiel, la bande passante en lecture est illustrée dans le fichier `task1_rw_bw_r.png` et celle en écriture dans `task1_rw_bw_w.png`. Pour l'accès aléatoire, la bande passante en lecture est représentée dans `task1_rand_rw_bw_r.png` et celle en écriture dans `task1_rand_rw_bw_w.png`. Les détails des graphiques sont les suivants : l'axe X indique les pourcentages d'écriture (de 0% à 100%), et l'axe Y montre la bande passante (en Ko/s ou Mo/s). Les données représentées incluent la bande passante moyenne, avec des barres d'erreur pour l'écart-type, la bande passante médiane sous forme de points individuels, ainsi qu'une ligne de régression linéaire indiquant la relation entre le pourcentage d'écriture et la bande passante.

## 6.2 Graphiques de la Tâche 2

Les graphiques de la tâche 2 montrent la bande passante en fonction de la taille de bloc pour les opérations de lecture et d'écriture. La bande passante en lecture est illustrée dans le fichier `task2_bw_req_size_r.png`, et celle en écriture dans `task2_bw_req_size_w.png`. Les détails des graphiques sont les suivants : l'axe X représente les tailles de blocs (de 1 Ko à 1 Mo) tracées sur une échelle logarithmique, et l'axe Y montre la bande passante. Les données représentées sont similaires à celles de la Tâche 1, avec un accent particulier sur l'impact de la taille de bloc sur les performances.

## 6.3 Graphiques de la Tâche 3

Les graphiques de la tâche 3 incluent une comparaison de la bande passante et une analyse de la latence. Pour la bande passante, les graphiques en lecture sont présentés dans `task3_bw_r.png` et en écriture dans `task3_bw_w.png`. L'analyse de la latence est représentée par un diagramme en barres dans `task3_latency_bar.png`, ainsi que des courbes de distribution cumulative (CDF) et des percentiles : la latence totale en lecture est dans `task3_lat_r_cdf_perc.png`, la latence totale en écriture dans `task3_lat_w_cdf_perc.png`, la latence de complétion en lecture dans `task3_clat_r_cdf_perc.png`, la latence de complétion en écriture dans `task3_clat_w_cdf_perc.png`, la latence de soumission en lecture dans `task3_slat_r_cdf_perc.png`, et la latence de soumission en écriture dans `task3_slat_w_cdf_perc.png`. Les détails des graphiques de bande passante incluent un axe X représentant les tailles de requêtes et l'étiquette de distribution mixte, et un axe Y montrant la bande passante. Les données représentent la comparaison des performances entre des tailles de blocs fixes et une distribution mixte de tailles de blocs. Le diagramme en barres de latence présente les latences totales, de complétion et de soumission pour les opérations de lecture et d'écriture, avec des latences moyennes et des barres d'erreur indiquant l'écart-type. Les graphiques CDF ont un axe X représentant la latence (en nanosecondes) et un axe Y pour la probabilité cumulative (%), montrant la distribution des valeurs de latence avec les percentiles marqués.

## 6.4 Graphiques de la Tâche 4

Les graphiques de la tâche 4 présentent la bande passante en fonction du nombre de travaux parallèles. La bande passante en lecture est illustrée dans le fichier `task4_numjobs_bw_r.png`, tandis que celle en écriture se trouve dans `task4_numjobs_bw_w.png`. Les détails des graphiques montrent que l'axe X représente le nombre de travaux parallèles (1, 2, 4, 6, 8) et l'axe Y indique la bande passante. Les données représentées sont similaires à celles des tâches précédentes, avec un accent particulier sur l'impact de l'augmentation du parallélisme sur les performances.

## 7 Analyse des Résultats Expérimentaux

### 7.1 Tâche 1 : Effet du Pourcentage d'Écriture sur la Bande Passante

#### 7.1.1 Observation

**Accès Séquentiel (rw)** À mesure que le pourcentage d'écriture augmente, la bande passante en lecture et en écriture diminue. (Graphiques : 1 montrent une tendance à la baisse.)

**Accès Aléatoire (randrw)** À mesure que le pourcentage d'écriture augmente, la bande passante en lecture et en écriture augmente. (Graphiques : 2 montrent une tendance à la hausse.)

#### 7.1.2 Analyse

Les tendances observées dans la Tâche 1 révèlent des comportements différents entre les schémas d'accès séquentiel et aléatoire lorsque le pourcentage d'écriture varie.

**Accès Séquentiel** Diminution de la Bande Passante avec l'Augmentation du Pourcentage d'Écriture :

Les opérations d'écriture séquentielle sont généralement plus lentes que les lectures séquentielles en raison du temps nécessaire pour écrire les données de manière sûre sur le support de stockage.

L'augmentation du pourcentage d'écriture signifie qu'une plus grande proportion d'opérations lentes est effectuée, ce qui réduit la bande passante globale.

De plus, les écritures peuvent impliquer des surcharges supplémentaires, comme la vidange du cache ou l'attente de l'accusé de réception du stockage, ce qui peut ralentir davantage les performances.

**Accès Aléatoire** Augmentation de la Bande Passante avec l'Augmentation du Pourcentage d'Écriture :

Les lectures aléatoires impliquent des déplacements fréquents des têtes de lecture (sur les disques durs), ce qui augmente la latence. Les écritures aléatoires peuvent parfois être optimisées par le système de stockage grâce à l'utilisation de caches ou de tampons d'écriture. Sur les supports de stockage modernes comme les SSD, les écritures aléatoires peuvent être traitées efficacement en raison de l'absence de pièces mobiles.

Ainsi, une augmentation du pourcentage d'écriture peut conduire à une meilleure utilisation des optimisations du système de stockage, augmentant la bande passante.

#### 7.1.3 Conclusion

Les tendances observées sont cohérentes avec les caractéristiques des opérations d'E/S.

La diminution de la bande passante en accès séquentiel avec plus d'écritures est attendue en raison de la nature plus lente des écritures séquentielles.

L'augmentation de la bande passante en accès aléatoire avec plus d'écritures peut être attribuée aux optimisations du système de stockage pour les écritures et à la latence inhérente des lectures aléatoires.

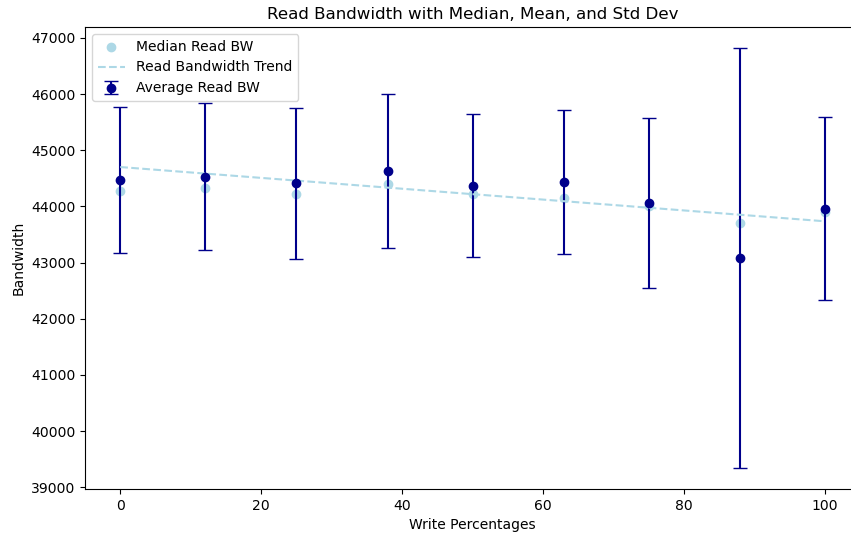
### 7.2 Tâche 2 : Effet de la Taille de Bloc sur la Bande Passante

#### 7.2.1 Observation

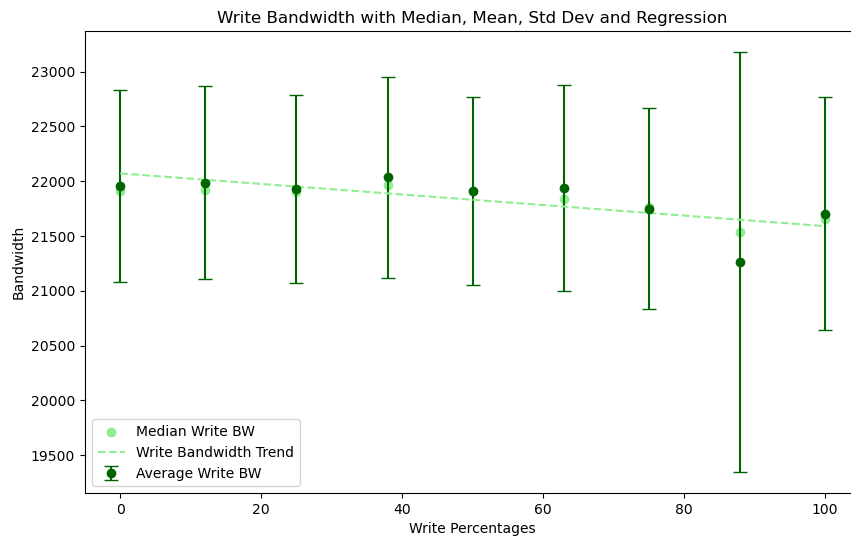
La bande passante en lecture et en écriture montre une légère tendance à la baisse avec l'augmentation de la taille de bloc. (Graphiques : 3)

#### 7.2.2 Analyse

**Grandes Tailles de Bloc** On pourrait s'attendre à ce que l'augmentation de la taille de bloc améliore la bande passante en réduisant le nombre total d'opérations d'E/S nécessaires.

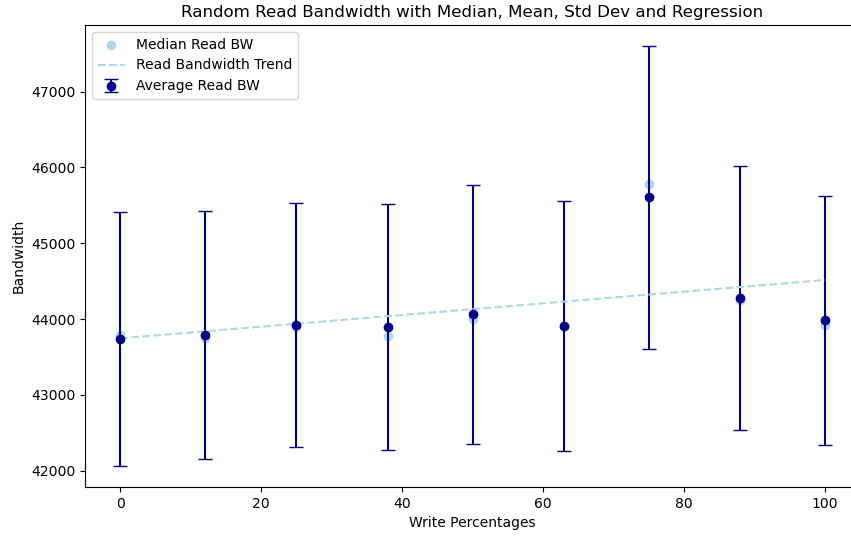


(a) Bande passante en lecture

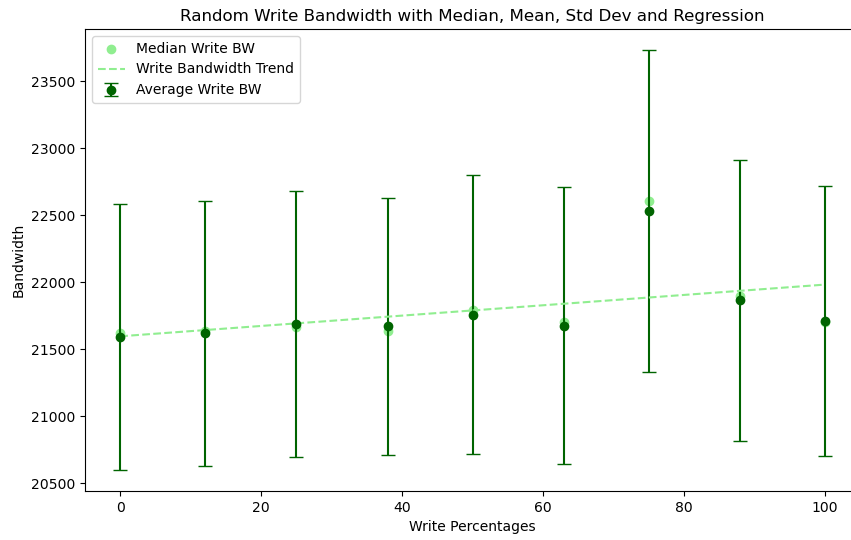


(b) Bande passante en écriture

FIGURE 1 – Bande passante en lecture et écriture à l'accès séquentiel

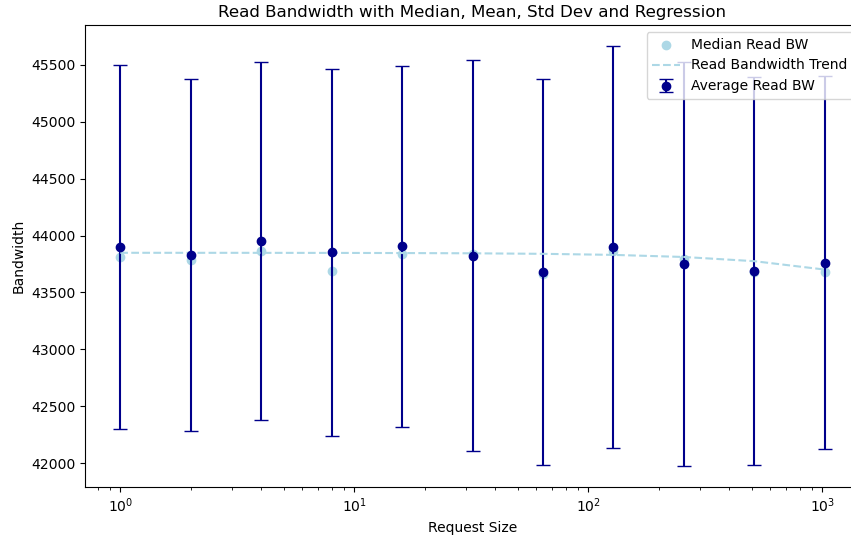


(a) Bande passante en lecture

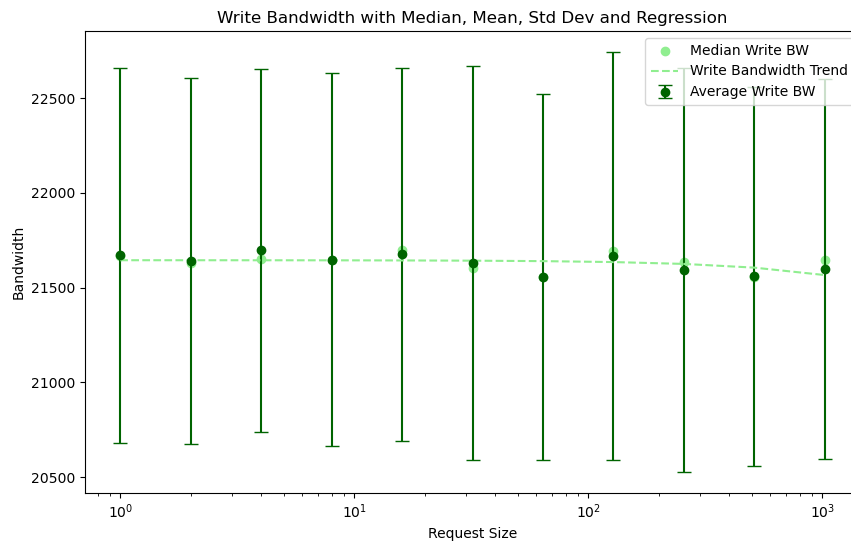


(b) Bande passante en écriture

FIGURE 2 – Bande passante en lecture et écriture à l'accès aléatoire



(a) Bande passante en lecture



(b) Bande passante en écriture

FIGURE 3 – Bande passante en lecture et écriture variée avec la taille de requêtes

Cependant, de grandes tailles de bloc peuvent entraîner des inefficacités, telles qu’une latence accrue par opération et une surcharge du système si le matériel sous-jacent ne gère pas efficacement les gros transferts.

Les caches et les tampons peuvent également ne pas être optimisés pour de grandes tailles de bloc, entraînant des performances réduites.

**Petites Tailles de Bloc** Les petites tailles de bloc augmentent le nombre d’opérations d’E/S, ce qui peut augmenter la surcharge du processeur et du système d’E/S.

Néanmoins, elles peuvent mieux s’adapter aux caches du système, améliorant potentiellement les performances.

### 7.2.3 Conclusion

La légère diminution de la bande passante avec l’augmentation de la taille de bloc est quelque peu inattendue. Cela pourrait s’expliquer par des limitations du matériel de stockage ou du système d’exploitation dans la gestion des grandes tailles de bloc. Il est possible qu’il existe une taille de bloc optimale pour ce système spécifique, et que dépasser cette taille n’apporte pas de bénéfices supplémentaires.

## 7.3 Tâche 4 : Effet du Nombre de Travaux sur la Bande Passante

### 7.3.1 Observation

La bande passante en lecture et en écriture augmente avec le nombre de travaux parallèles. (Graphiques : 4)

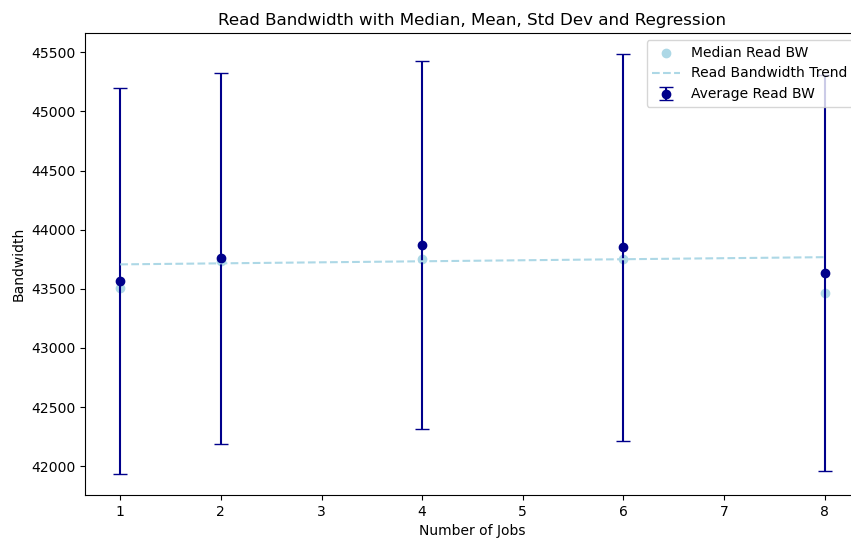
### 7.3.2 Analyse

L’augmentation du nombre de travaux parallèles permet de mieux exploiter le parallélisme intrinsèque des supports de stockage modernes, en particulier les SSD. Un plus grand nombre de requêtes simultanées peut améliorer la saturation du bus d’E/S et permettre au système de stockage d’optimiser la planification des opérations. Cela conduit à une meilleure utilisation des ressources et à une augmentation de la bande passante globale.

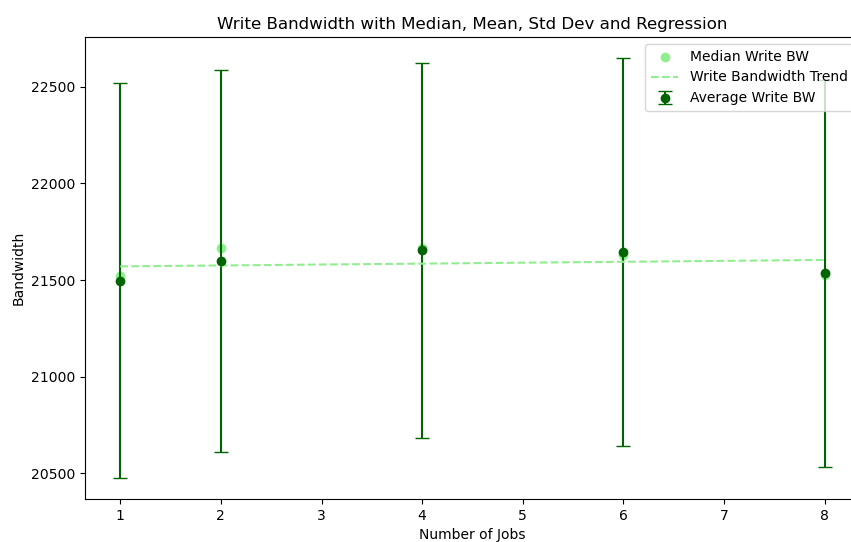
### 7.3.3 Conclusion

L’augmentation de la bande passante avec le nombre de travaux est un résultat attendu. Cela reflète la capacité du système à tirer parti du parallélisme pour améliorer les performances. Cependant, il faut noter qu’au-delà d’un certain point, l’ajout de plus de travaux peut ne plus apporter de gains significatifs en raison de la saturation du matériel ou d’autres goulots d’étranglement.





(a) Bande passante en lecture



(b) Bande passante en écriture

FIGURE 4 – Bande passante en lecture et écriture variée avec le nombre de travaux

## 7.4 Tâche 3 : Analyse de la Bande Passante

### 7.4.1 Observation

Pour lecture et écriture, la bande passante des tailles de blocs mixtes (30% 4 Ko, 60% 16 Ko et 10% 64 Ko) est diminuée d'environ 20% - 25% de celle des tailles de blocs fixes (100% 4 Ko, 100% 16 Ko ou 100% 64 Ko). (Graphiques : 5)

### 7.4.2 Analyse

Lorsqu'on utilise des tailles de blocs mixtes, les données sont fragmentées en différents formats, ce qui entraîne une surcharge supplémentaire pour le système. Les opérations d'entrée/sortie doivent gérer différentes tailles de blocs, ce qui peut augmenter la complexité du traitement des données. Cette fragmentation peut engendrer des temps de latence plus longs, car le système doit gérer plusieurs tailles de requêtes.

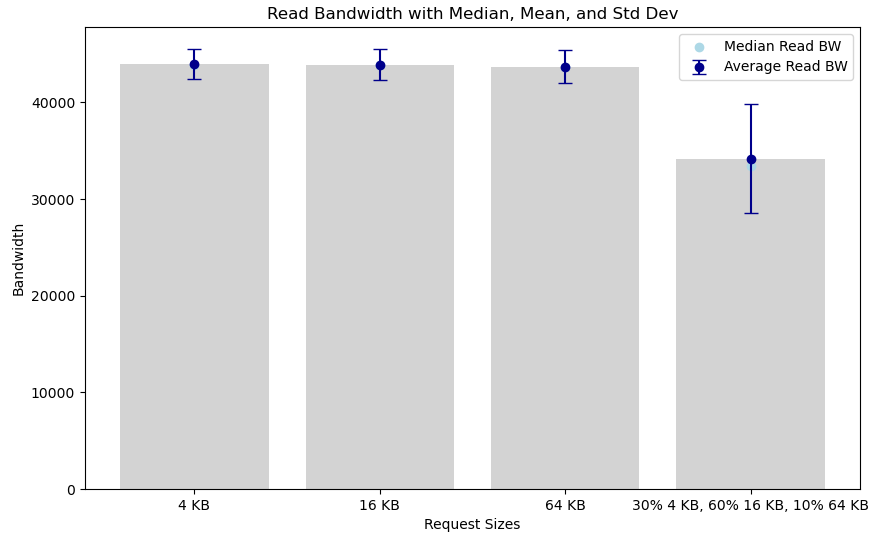
Le système de fichiers et le matériel sous-jacent doivent traiter les requêtes d'écriture et de lecture pour des tailles de blocs variées. Cela implique une gestion supplémentaire des métadonnées et une planification des opérations d'E/S. Cette surcharge peut réduire l'efficacité globale des opérations, car le temps consacré à gérer les tailles de blocs mixtes peut dépasser celui nécessaire pour traiter des tailles de blocs fixes.

Les systèmes de stockage sont souvent optimisés pour des tailles de blocs spécifiques. Par exemple, un disque dur ou un SSD peut être conçu pour traiter efficacement des blocs de 4 Ko ou 16 Ko. Lorsque des tailles de blocs mixtes sont utilisées, le matériel peut ne pas fonctionner à son niveau optimal, entraînant une réduction de la bande passante. Les performances peuvent diminuer si le matériel doit jongler entre des tailles de blocs qui ne correspondent pas à son design optimal.

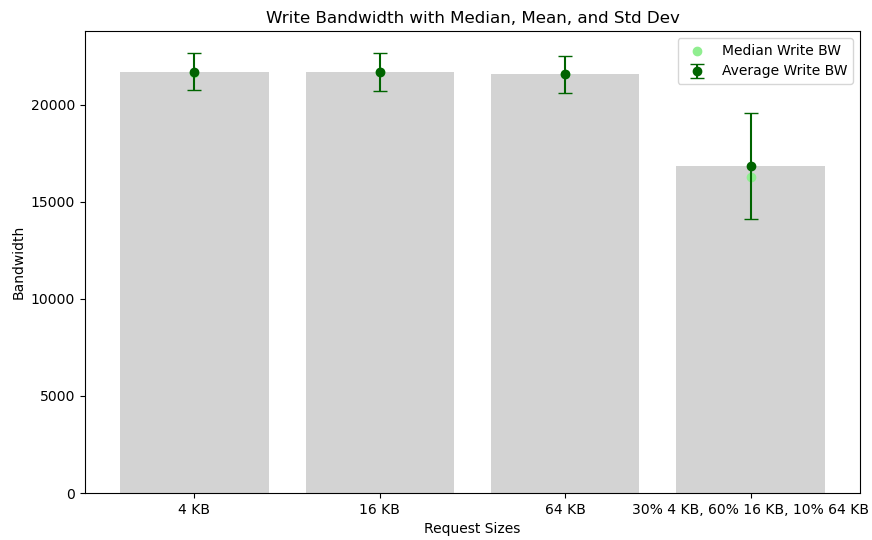
Les tailles de blocs mixtes peuvent augmenter la latence des opérations d'E/S. Par exemple, les demandes de blocs de 64 Ko peuvent introduire des délais plus importants par rapport à celles de 4 Ko et 16 Ko. Cette latence accrue, qui résulte de la gestion des différents types de blocs, peut contribuer à une bande passante globale plus faible.

### 7.4.3 Conclusion

La baisse de 20% à 25% de la bande passante lors de l'utilisation de tailles de blocs mixtes par rapport à des tailles de blocs fixes est le résultat d'une combinaison de fragmentation des données, de surcharge de gestion, d'optimisation matérielle et d'augmentation de la latence. Ces facteurs se cumulent pour diminuer l'efficacité globale des opérations d'entrée/sortie, conduisant ainsi à une performance inférieure dans le cadre de la manipulation de tailles de blocs mixtes.



(a) Bande passante en lecture



(b) Bande passante en écriture

FIGURE 5 – Bande passante en lecture et écriture : tailles de requêtes fixes et taille de requête mixte

## 7.5 Tâche 3 : Analyse de la Latence

### 7.5.1 Observation

**Histogramme** Pour lecture et écriture, la latence totale est à peu près la somme de la latence de complétion et celle de soumission. Pour certaines machines, la latence de complétion domine, pour les autres, c'est la latence de soumission qui domine. (Graphiques : les résultats de cette expérience : 6 les résultats d'une autre machine 7)

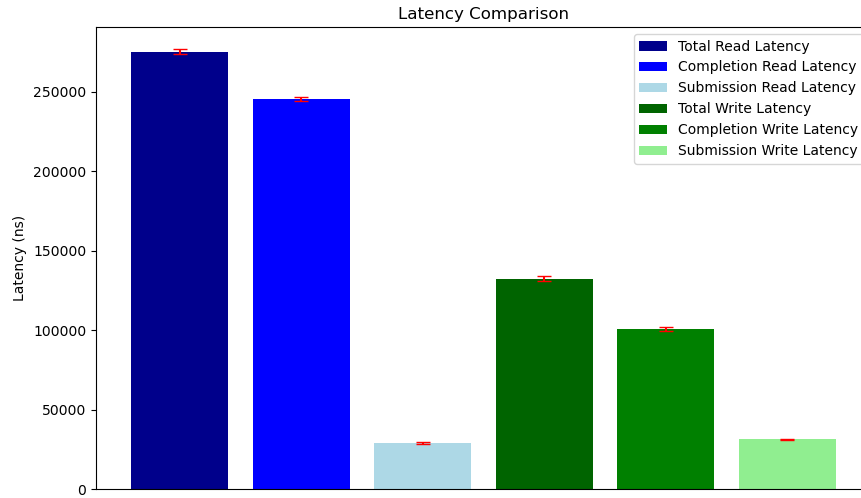


FIGURE 6 – Histogramme des Latences

**Fonction de répartition cumulative et percentiles** Les graphiques de la fonction de répartition cumulative (CDF) sont strictement croissants. Sur chaque graphique, les points de percentiles se situent assez proches de la courbe CDF mais ils ne sont pas superposés. (Graphiques : 8 9 10 )

### 7.5.2 Analyse

Dans cette étude, nous avons examiné les latences dans les différentes phases des opérations d'entrée/sortie : la latence totale représente le temps total écoulé entre la soumission de la requête et sa complétion, la latence de complétion est le temps pris par le système de stockage pour traiter la requête après sa prise en charge jusqu'à sa complétion, et la latence de soumission correspond au temps nécessaire pour soumettre la requête au système de stockage. Les analyses des latences permettent d'identifier les éventuels problèmes de performances, comme des latences élevées ou des variations importantes.

**Histogramme** Pour cette machine (qui est installée sur une clé USB), la latence de complétion est plus importante, cela signifie que le stockage prend plus de temps à traiter chaque requête. La latence totale en lecture est plus importante que celle en écriture, et les deux sont plus élevées que celles de l'autre machine, surtout la latence en lecture, qui est plus de deux fois plus élevée. Cela signifie que l'optimisation de lecture n'est pas suffisante sur cette machine.

**CDF et Percentiles** Les CDF représentent la probabilité qu'une variable aléatoire soit inférieure ou égale à une certaine valeur. Une tendance croissante est normale, indiquant que les latences mesurées s'accumulent

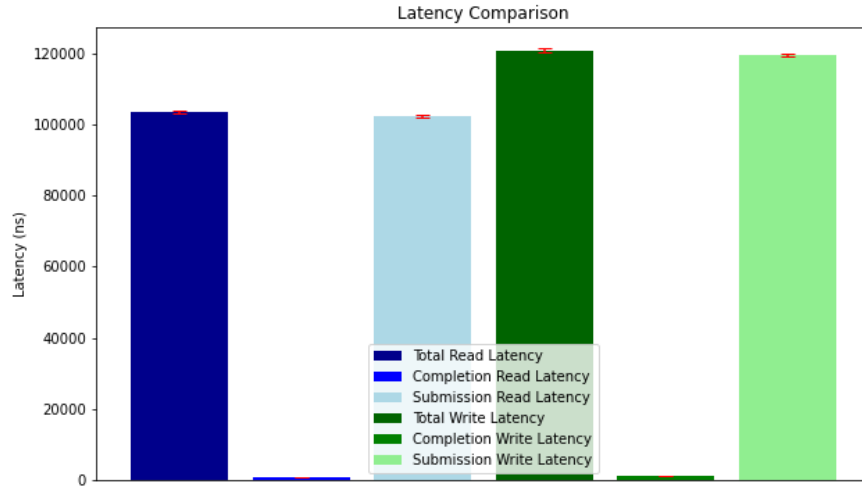
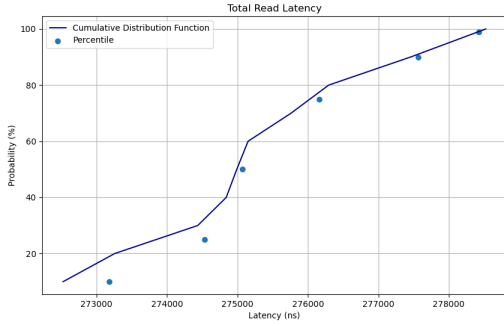
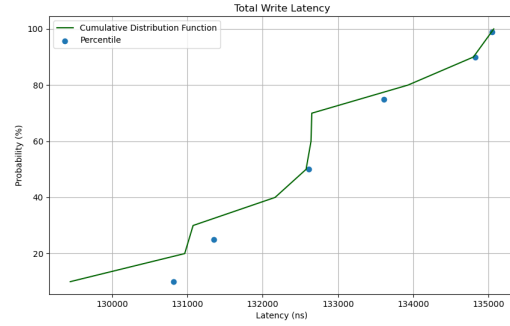


FIGURE 7 – Histogramme des Latences pour une autre machine

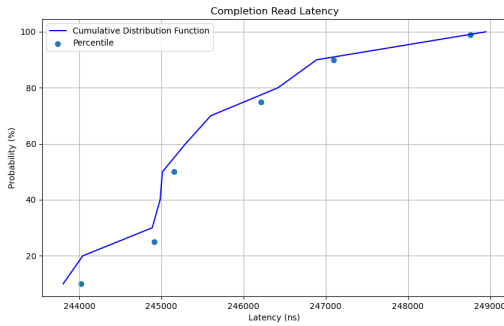


(a) Latence totale en lecture

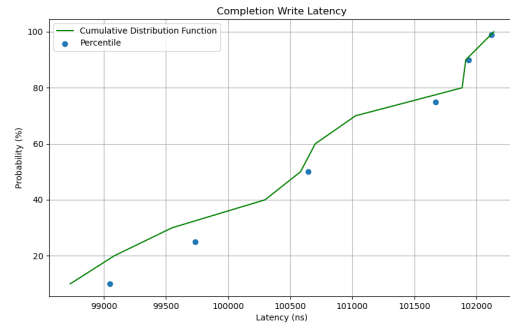


(b) Latence totale en écriture

FIGURE 8 – CDF et Percentiles pour la Latence totale

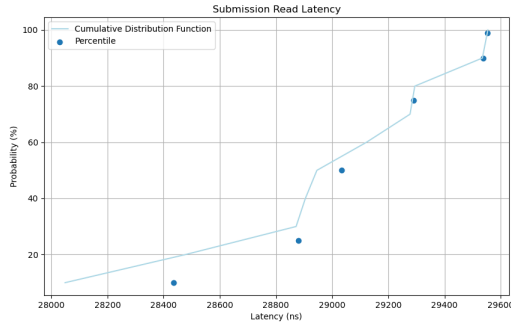


(a) Latence de Complétion en lecture

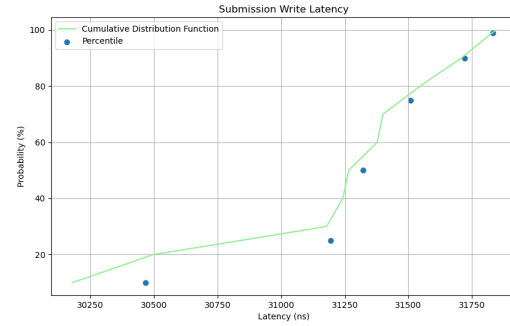


(b) Latence de Complétion en écriture

FIGURE 9 – CDF et Percentiles pour la Latence de Complétion



(a) Latence de Soumission en lecture



(b) Latence de Soumission en écriture

FIGURE 10 – CDF et Percentiles pour la Latence de Soumission

au fur et à mesure que la valeur augmente. Cela permet d'analyser la distribution des latences et d'identifier les valeurs seuils pour différents percentiles (ici : 10%, 25%, 50%, 75%, 90%, 99%).

### 7.5.3 Conclusion

Les tendances croissantes des CDF de latence sont normales et attendues. Elles indiquent que la majorité des opérations se terminent dans des délais raisonnables. Si des anomalies ou des latences élevées sont observées, cela pourrait indiquer des problèmes à investiguer plus en détail.

## 8 Conclusion

Les résultats de ces expériences sont majoritairement raisonnables et reflètent les comportements attendus des systèmes d'E/S sous différentes configurations. Les analyses ont permis d'identifier et d'expliquer les tendances observées, tout en soulignant les domaines nécessitant une attention particulière ou des investigations supplémentaires.

### 8.1 Caractère Raisonnable des Résultats

Dans l'ensemble, les résultats observés sont cohérents avec les connaissances sur les systèmes d'E/S. Les tendances correspondent aux comportements attendus des opérations de lecture et d'écriture dans différents scénarios.

### 8.2 Causes Possibles pour des Résultats Non Raisonables

Les limitations matérielles peuvent affecter les performances de stockage, car le matériel peut présenter des variations en fonction des configurations, tandis que les surcharges système dues à d'autres processus ou à une utilisation élevée du CPU peuvent également altérer les résultats ; de plus, des effets de cache mal gérés peuvent influencer les performances mesurées, et des variations dans les conditions de test peuvent introduire des incohérences dans les résultats obtenus.

### 8.3 Améliorations possibles

Il est essentiel de vérifier que les tests sont effectués en I/O direct pour minimiser l'impact du cache, de s'assurer que le système est dans un état stable sans charge significative pouvant affecter les mesures, et de réaliser des tests supplémentaires ou d'ajuster les paramètres afin d'approfondir l'analyse des résultats inattendus.