

Guide pour la présentation de résultats

La présentation des résultats est un élément essentiel d'un compte-rendu scientifique. L'objectif est de montrer de la manière la plus claire possible la pertinence des travaux réalisés et, par conséquent, la qualité du travail. Lorsque cette partie est insuffisamment préparée, on court le risque d'influer de manière négative, indépendamment de la qualité intrinsèque du travail.

Dans le cadre du développement d'algorithmes pour les problèmes d'aide à la décision, les critères de pertinence comprennent en particulier :

- la validité des résultats : est-ce que les solutions sont correctes pour le problème posé ?
- la qualité des résultats : est-ce que les solutions sont de bonne qualité ?
- les temps de calcul : est-ce que le temps nécessaire est en adéquation ?

En règle générale, la validité est souvent considérée comme allant de soi. Il suffit d'écrire une routine de vérification durant la phase de débogage et de lui soumettre toutes les solutions produites. Lorsque plus aucune solution invalide n'est identifiée, on peut alors passer à la phase de test et limiter l'appel de cette fonction à la solution retournée par l'algorithme. Ceci n'implique évidemment pas l'absence de bugs dans le code !

Pour juger de la qualité des solutions, il y a plusieurs cas possibles :

- ce sont les premiers travaux publiés sur le problème. En l'absence de référentiel de comparaison, la seule base dont on dispose sont les solutions produites par les différents algorithmes testés. Si les algorithmes ne sont pas exacts (heuristiques et métaheuristiques) on ne peut pas juger de la qualité absolue des solutions. On peut juste juger de la qualité relative en comparant les solutions entre elles. Attention alors au piège qui consiste à généraliser une conclusion à partir de l'analyse sur deux ou trois instances.
- d'autres résultats existent et ce sont des méthodes approchées. Comme précédemment, on ne pourra pas tester la qualité absolue des solutions, juste la qualité relative.
- d'autres résultats existent et on a les solutions optimales. Dans ce cas, on pourra aussi juger de la qualité absolue des solutions.

Pour comparer la valeur f_1 et f_2 de deux solutions, on évite surtout la comparaison absolue, c'est à dire la mesure de la différence absolue $f_1 - f_2$. La raison est simple : si on multiplie la fonction objectif par 100, la différence absolue est multipliée par 100 aussi. On est donc dépendant du facteur d'échelle, ce qui complique les choses. Il vaut mieux effectuer une comparaison relative $(f_1 - f_2) / f_1$. De la sorte, on s'affranchit du facteur d'échelle et la valeur indique un pourcentage.

Dans le cas du PDPTW, la fonction objectif contient deux critères : le nombre de véhicules (critère prioritaire, à minimiser) et la distance totale (critère secondaire, à minimiser). Dans les tableaux de résultats, on doit donc reporter les deux valeurs correspondant à la solution. La comparaison entre solutions se fait critère-à-critère.

L'organisation de la section des résultats est souvent la suivante :

- on commence par présenter le contexte expérimental : type de processeur, quantité de mémoire vive, type de système d'exploitation, langage de programmation et compilateur. On mentionne aussi les bibliothèques ou les logiciels annexes qui ont été éventuellement utilisés pour produire les résultats.
- On présente ensuite la méthodologie d'expérimentation et en particulier les réglages sur chaque algorithme (nombre maximal d'itérations, limite de temps de calcul, etc.). On précise le protocole si nécessaire. Par exemple, pour les méthodes utilisant l'aléatoire, on voit souvent dans la littérature des résultats qui sont les meilleurs obtenus sur 5 ou 10 runs de la méthode.
- Ensuite, si cela vaut le coup, on présente les résultats spécifiques à chaque méthode. C'est

souvent fait pour justifier certains choix. Par exemple, pour justifier l'utilisation de certaines structures de voisinages, de certaines méthodes de génération de solution initiale, de certaines méthodes de gestion du processus de recherche dans l'espace des solutions.

- Enfin on présente les résultats comparant les méthodes entre elles. C'est un point délicat car on doit présenter les résultats sur toutes les instances et pour toutes les méthodes. Du coup, le tableau devient facilement indigeste. C'est pour ça que si on a plusieurs variantes d'un même algorithme, on place les comparaisons dans un tableau préliminaire. Pour garder un peu de légèreté, on ne met que l'essentiel. Chaque ligne correspond à une instance. On commence par donner le nom de l'instance et ses grandeurs caractéristiques (pour le PDPTW c'est le nombre de requêtes). Ensuite on présente les résultats de chaque méthode. D'abord les résultats des méthodes existant dans la littérature, s'ils existent. Puis nos résultats. Se limiter d'abord à temps de calcul et valeur de la solution. S'il y a de la place, on peut ajouter une colonne pour chaque méthode donnant l'écart à la meilleure solution de la littérature si elle existe.
- Les tableaux de résultats doivent toujours être présentés et surtout ils doivent toujours être suivis d'une analyse ou de commentaires. Ces commentaires peuvent aussi passer par l'ajout de diagrammes ou de tableaux annexes. Par exemple pour donner des statistiques comme le nombre de fois où telle ou telle méthode donne le meilleur résultat. Ou encore, pour une instance donnée et une métaheuristique donnée, l'évolution itération par itération de la valeur de la meilleure solution. Ou encore une analyse supplémentaire pour comparer les méthodes de manière plus équitable. En effet, il peut être difficile de comparer des méthodes ayant des temps de calculs complètement différents. On peut alors faire un test supplémentaire consistant à fixer le même temps de calcul pour chaque méthode et regarder la valeur produite par chaque méthode. Réciproquement, on peut fixer un critère d'arrêt sur la valeur de solution à obtenir et regarder le temps nécessaire à chaque méthode pour y parvenir.

Dans tous les cas, le point important est de se dire que le lecteur ne sait pas ce qu'on a fait. Il faut donc en permanence se demander ce qu'on veut transmettre comme message au lecteur et si on est suffisamment clair.