

1 Théorie de la complexité

La théorie de la complexité est un domaine qui va permettre de quantifier les ressources nécessaires à la résolution d'un problème de manière mathématiques. L'algorithme va permettre de réaliser ce problème posé. Cependant un problème a plusieurs algorithmes répondant à celui-ci. Pour savoir lequel sera le plus efficace, il faut compter le nombre d'étapes nécessaires à la résolution.

Il existe deux problèmes d'algorithmie :

- Les problèmes de décision par "oui" ou par "non"
- Les problèmes de conditions pour la recherche d'une solution

Les modèles de calculs utilisés pour la théorie de la complexité sont les machines déterministe et la machine non déterministe.

La machine déterministe possède une seule action possible en fonction de l'état dans lequel il se trouve à un instant t .

La machine non déterministe quand à elle possède plusieurs actions réalisables à un instant donné.

La machine représentant ce concept est la machine de Turing. Cette machine est l'ancêtre de la mémoire de l'ordinateur.

Elle est constituée d'une bande divisée en cases et d'une tête pour lire et écrire sur la case, la mémorisation de l'état en cours de la machine. Chaque case contient un symbole d'un alphabet qui est généralement fini. Les cases vont se déplacer sur la droite ou sur la gauche. De base, cette machine va permettre de réaliser des additions et des soustractions. Cette bande doit cependant être théoriquement infinie ce qui correspond à avoir un espace de stockage infinie. Les machines d'aujourd'hui sont des machines de Turing équivalentes du fait que l'on n'est pas de mémoire infinie. Une telle machine serait dite Turing complet.

La machine équivalente à celle-ci aujourd'hui est la RAM que l'on retrouve sur nos ordinateurs.

Il existe différentes classes de complexité :

- Les problèmes de classe L sont des problèmes qui vont être réalisables par une machine de Turing déterministe et dont l'espace mémoire utilisé est de taille logarithmique par rapport à l'entrée.

- Les problèmes de classe NL sont quand à eux réalisable par une machine de Turing non déterministe
- Les problèmes de classe P sont des problèmes de décisions faisable en un temps polynomial par rapport à l'entrée. Ils sont réalisable avec une machine déterministe.
- Les problèmes de classe NP sont aussi des problèmes de décisions à temps polynomial sur une machine non déterministe. Ces problèmes vont être réalisable en listant l'ensemble des solutions et en les testants. Mais ceci prend énormément de temps selon ce que l'on a en entrée.
- Les problèmes dit NP complet sont des problèmes au minimum aussi difficile que les problèmes NP. On peut trouver des solutions à ces problèmes, mais pas de manière efficaces. Le temps pour résoudre ces problèmes sont de l'ordre exponentiel par rapport à l'entrée. Savoir si il existe un algorithme polynomial pour résoudre un problème de type NP complet permettrait de dire que P est égal NP. Et dans le cas contraire dire que P est différent NP. Ceci fait partie des questions non résolues les plus importante à ce jour.

Parler des autres classes de problème au dessus

2 Problème d'ordonnancement

Le but de l'ordonnancement est d'organiser des taches de manière optimales. Celui est accompagné de contrainte de type temporelle, la disponibilité et les ressources.

Dans le cas de l'emploi du temps, les problèmes temporelles vont être au niveau de la durée des cours. Un cours doit durer 2h ou 4H et ceci sans interruptions (le midi ou la fin de journée). De plus des délais doivent être respectés vis-a-vis des examens. Au niveau des disponibilités, le problème sera au niveau des enseignants, à savoir quand ils peuvent être libéré de leurs travaux pour les consultants et du coup optimiser leurs journée de libre. Enfin pour les ressources, il va s'agir des classes de cours avec un nombre de place restreint dans chacune des classes, mais aussi des labos en fonction des travaux pratiques réalisé.

Les taches de cette ordonnancement vont être les cours. Ils vont avoir une date, une heure de début et de fin et vont être devoir réalisé dans interruptions dans la journée. Les heures de cours vont être de 2 ou 4 heures. La journée est divisée en

deux créneaux de 4H, le matin et l'après midi. Enfin chacun des cours doit être réalisé sans interruptions.

Les ressources vont être les classes de cours et les enseignants. Les classes ont un certain nombre de place et les laboratoires ont des caractéristiques différentes dans la réalisation d'un TP. De plus des salles de cours se trouvent dans différentes villes. Ainsi une classe ne peut faire un cours de 2H dans une certaine ville puis les 2H qui suivent dans une autre (dans le cas où les deux villes sont très proches, accessible en transport en commun comme le métro ou le RER). Pour une ville éloignée, les cours devront se dérouler exclusivement dans cette même ville. Enfin une salle ne peut contenir qu'un cours à la fois.

Au niveau des enseignants nous retrouvons les mêmes contraintes que pour les salles de cours. Mais ajouté à cela, ils ont des horaires de disponibilité par semaine. Pour les consultants qui vont être présents de manière occasionnelle, ils ne vont pouvoir être disponible qu'un certain nombre de jour bien précis en fonction de leurs travaux en entreprise, c'est pourquoi ils vont être prioritaire sur les enseignants de l'école.

Enfin nous allons trouver des contraintes par rapport aux examens. Ils sont définis à une date précise et de ce fait, le programme des cours devra être fini avant le début des examens.

Deux types de réponses vont être acceptés pour l'ordonnancement, optimisé la solution, ou le fait qu'elle soit acceptable. Une réponse optimale prendrait énormément de temps à être réalisée. C'est pourquoi l'objectif va être d'avoir une des meilleures solutions qui arrive à concilier les différentes contraintes entre elles.

3 Méthodes de résolutions

3.1 Heuristiques

L'heuristique est un système de calcul qui va permettre non pas de trouver une réponse optimisée à un problème mais une solution qui va pouvoir être acceptable de qualité plus rapidement. Elle utilise une technique empirique. En fonction de l'état dans lequel on est actuellement, on va en déduire une stratégie pour la suite.

3.2 Programmation linéaire

Dans la programmation linéaire, les contraintes et les objectifs vont être des fonctions linéaire.

Je ne sais pas comment on va linéariser la disponibilité des profs

3.3 Programmation par contrainte

La programmation par contrainte va permettre de trouver une solution à un problème avec des grandes combinatoire comme l'ordonnancement. Elle est modélisé par des décisions et des contraintes sous forme de variable. La recherche de solution va passer par une accumulation des contraintes pour limiter le nombre de solution possible. Ceci va nous permettre de savoir aussi si il existe une solution ou non à notre problème en fonction des différentes contraintes que nous allons avoir.

Pour la recherche d'une solution il est possible de lister l'ensemble des solutions possibles puis de les tester pour savoir si oui ou non elles répondent aux contraintes qui ont été imposé. Cependant cette technique ne peut s'appliquer que sur des problèmes de petites tailles. Dans les autres il y aurait beaucoup trop de solution possible à énumérer puis à tester.

Pour les problèmes de plus grosses envergures, il va falloir utilisé un filtrage. L'objectif est de trouver les valeurs impossibles avec les contraintes qui nous ont été imposé. Il est nécessaire de découper notre problèmes avec les différentes variables et de faire le filtrage sur chacune d'entre elles jusqu'à trouver une solution impossible.

Le solveur que nous allons utilisé est Prolog.
Il va fonctionner avec :

des prédicats Il s'agit d'une variable avec un argument et lors d'une recherche on va vérifier si ce que l'on a est vrai par rapport à cette variable ou non.

des règles C'est un ensemble logique de fait. Si quelque chose est vrai et qu'un autre point aussi est vrai, alors il s'agit ça.

l'évaluation On un esemble de règles qui vont nous permettre de répondre à une question par oui ou non.