

Université du Québec à Chicoutimi
8INF926 - Atelier en optimisation avancée
Projet 1 (15%)

Hiver 2024

Professeure : Sara Séguin

Cette première portion du projet comporte 2 parties. Dans un premier temps, vous aurez à modéliser des fonctions de production à l'aide de Matlab, puis vous aurez à implémenter un algorithme de programmation dynamique.

1. Modélisation des fonctions de production à partir de données réelles

Un fichier contenant des données réelles vous est remis. Ce dernier contient les données réelles de production pour une centrale, de 2013 à 2017, aux 2 minutes. La centrale comporte cinq turbines. L'en-tête du fichier est le suivant :

- Date
- Elav : élévation aval
- Qtot : Débit total turbiné à la centrale
- Qvan : Débit total déversé à la centrale
- Niv Amont : élévation amont
- Q1,Q2,Q3,Q4,Q5 : débit turbiné par chaque turbine
- P1,P2,P3,P4,P5 : puissance produite par chaque turbine.

Lors de la deuxième partie du projet, vous implémenterez un algorithme de programmation dynamique visant à maximiser l'énergie totale produite. Or, vous n'aurez pas les représentations analytiques des fonctions de production, seulement les données contenues dans ce fichier.

Pour ce projet, vous devez obtenir une expression analytique des fonctions de production pour chaque turbine, afin de les utiliser dans le deuxième projet.

La puissance produite par une turbine dépend de la hauteur de chute nette et du débit turbiné. Or, la hauteur de chute nette dépend aussi du débit total, puisque cela fait varier l'élévation aval. Ainsi, vous devez modéliser :

- L'élévation avale en fonction du débit total
- La puissance produite en fonction de la hauteur de chute nette et du débit turbiné, et ce, pour chaque turbine.

Pour ce faire, vous devrez calculer la hauteur de chute nette à partir des données réelles. Pour les pertes, utilisez $0.5 \times 10^{-5} \times Q_{unit}^2$.

Utilisez le toolbox de curve fitting de Matlab pour obtenir les représentations analytiques des fonctions à approximer.

Votre travail consiste à obtenir une bonne représentation analytique des fonctions de production et de l'élévation aval. Vous devez justifier vos choix ainsi que votre démarche. Par exemple :

- Choix du degré, pourquoi ?
- Erreur
- Visualisation
- Pré-traitement des données, justifications
- etc.

Partie II : Programmation dynamique

Dans le domaine de la gestion de la production hydroélectrique, plusieurs types de modèles d'optimisation sont utilisés afin d'assurer une gestion efficace du réseau. Dans le cadre de ce projet, nous développons un outil de gestion aux centrales hydroélectriques. Cet outil doit déterminer les turbines en marche et le débit qu'elles doivent turbiner afin de maximiser la puissance totale produite. C'est un outil primordial pour les producteurs hydroélectriques car l'efficacité des groupes turbo-alternateurs varie d'un groupe à l'autre. Pour une consigne de débit total, il existe une répartition optimale maximisant la puissance totale.

Tel que présenté en cours, la puissance unitaire produite par un groupe turbo-alternateur dépend du débit unitaire mais aussi de la hauteur de chute unitaire. Il sera nécessaire d'inclure un module permettant le calcul de ces intrants. De plus, des restrictions d'opération peuvent être appliquées lors de l'optimisation. En effet, certains groupes peuvent ne pas être disponibles ou encore être restreints à opérer dans une plage spécifique. Il faudra donc en tenir compte dans votre calcul et votre interface graphique.

Cette partie consiste à développer et implémenter un algorithme de programmation dynamique pour le chargement optimal des groupes. Selon un débit total à répartir et une élévation amont donnée, vous devez répartir l'eau entre les différentes turbines afin de maximiser la puissance produite.

Vous devez d'abord modéliser mathématiquement le problème d'optimisation. Soyez consistants dans votre notation. Votre modèle doit maximiser la puissance produite, selon un débit total à répartir et une élévation amont donnée. Votre algorithme doit déterminer la meilleure façon de répartir le débit total entre les différentes turbines. Vous devez utiliser les fonctions de production de chaque turbine et la fonction de l'élévation aval en fonction du débit total développées lors du projet 1.

Vous devez considérer qu'une turbine peut être indisponible (débit pouvant être turbiné de 0) ou qu'une limitation de débit maximale soit imposée (inférieure au débit maximal habituel).

Par la suite, vous devez implémenter l'algorithme dans un système informatique. Vous pouvez utiliser Python, C++ ou Julia. Gardez en tête que la partie III du projet est la conception d'une interface utilisateur. Commentez votre code.

Validation des résultats

Afin de valider les résultats, testez sur les 100 premières valeurs (débit total et élévation amont) du fichier EXCEL contenant les données initiales. Vous devez comparer les résultats obtenus avec votre algorithme et les données du fichier excel. Par exemple, comparez la puissance produite par votre algorithme et la puissance dans le fichier excel. Si une turbine est indisponible dans le fichier excel, elle doit aussi l'être dans votre algorithme !

Utilisez les caractéristiques suivantes :

- Débit minimal par turbine = $0 \text{ m}^3/s$
- Débit maximal par turbine = $160 \text{ m}^3/s$
- Discrétisations du débit = $5 \text{ m}^3/s$

Vous devez modéliser mathématiquement votre modèle de programmation dynamique. Bien définir les étapes, les états, les variables de décision, le déroulement de l'algorithme.

Noter aussi le temps de résolution de votre algorithme de programmation dynamique.

Rapport

Vous devez remettre un rapport complet avec page-titre comprenant :

- Introduction (1 à 2 pages). Mise en contexte du projet, explication du devoir.
- PARTIE 1 :
- Explication des méthodes de modélisation pertinentes de fonctions (2 à 3 pages).
- Méthodologie retenue dans le cadre de votre projet pour la partie 1 (1 à 2 pages). Librairie matlab, choix du degré, erreurs, visualisation des données, pré-traitement, etc. Méthodologie retenue dans le cadre de votre projet pour la partie 2 (1 à 2 pages). Librairie Julia, présentation du modèle mathématique puis présentation de l'implémentation (fonction objectif, variables, paramètres, etc.).
- Résultats partie 1 (les fonctions) ainsi que des figures montrant les nuages de points avec vos fonctions. Résultats partie 2. Temps requis pour l'optimisation. Solution (valeurs des variables et de la fonction objectif) (nombre de pages nécessaires)
- PARTIE 2 :
- Explication de la programmation dynamique (2 à 3 pages)
- Méthodologie, plus précisément la modélisation mathématique de votre problème de chargement optimal, le choix du langage de programmation et du logiciel, etc. Attention à la notation mathématique. (1 à 2 pages)
- Résultats. Comparez les résultats des 100 premières lignes données du fichier et ceux obtenus par votre algorithme de programmation dynamique, temps de résolution, effet de la modélisation en partie I sur la précision des résultats obtenus. (nécessaire)
- Discussion (retour sur résultats, difficultés rencontrées, hypothèses requises pour compléter

- le travail, etc). (1 page)
- Conclusion et travaux futurs. (1/2 page)
- Bibliographie.

ATTENTION aux éléments suivants

- Table des matières
- Toutes les figures et tableaux doivent être identifiés, numérotés et cités dans le corps du texte.
- L'identifiant d'une figure est placé SOUS la figure.
- L'identifiant d'un tableau est placé AU-DESSUS de la figure.
- CITEZ VOS SOURCES. Voir le site de la bibliothèque pour plus de détails.
- Numéros de page