

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Abderrahmane Mira de Béjaïa
Faculté des Sciences Exactes
Département de Recherche Opérationnelle



En vue de l'obtention du diplôme de Master en Recherche Opérationnelle

Option : Modélisation Mathématique et Techniques de Décision

Thème

*Sur l'optimisation dans le système d'attente
GI/M/1/N avec "working vacation"*

Présenté par :

Mlle AZIB Saida et *Mme* Mahdjate Zahia née Djerroud

Devant le jury composé de :

Président	<i>M. K. Abbas</i>	Professeur	U. Béjaïa
Rapporteur	<i>M. S. Ouazine</i>	M.C.B	U. Béjaïa
Éxaminatrice	<i>Mme B. Barache</i>	M.C.B	U. Béjaïa
Éxamineur	<i>M. M. Soufit</i>	Doctorant	U. Béjaïa

Promotion : 2017-2018

Remerciements

Au terme de ce travail, nous tenons à exprimer dans ces lignes notre reconnaissance à toutes les personnes qui ont participé d'une manière ou d'une autre à son aboutissement.

En premier lieu, nous tenons à exprimer notre profonde gratitude à M. OUAZINE SOUFIANE pour sa disponibilité, ses précieux conseils et le soutien qu'il nous a apporté pendant la réalisation de notre travail. Il n'a jamais manqué de nous conseiller et de nous orienter tout au long de toute la période de réalisation de notre mémoire. Qu'il trouve ici l'expression de notre respect et de notre profonde reconnaissance.

Nous présentons nos sincères remerciements aux membres du jury qui ont accepté d'évaluer notre travail.

Nous voudrions exprimer notre profonde reconnaissance et remerciements à nos parents qui ont tant sacrifié dans leurs vies, pour nous garantir une éducation et un avenir.

Enfin, nous tenons à remercier nos familles respectives et nos amis pour leur aide et leur soutien moral.

Table des matières

Introduction générale	5
1 Systèmes de file d'attente	7
1.1 Chaîne de Markov finie	7
1.2 Description du phénomène d'attente	10
1.2.1 Types de files d'attente	12
1.2.2 Disciplines de service	13
1.2.3 Notation de Kendall	14
1.2.4 Analyse économique de la théorie des files d'attente	14
1.2.5 Analyse mathématique d'un système de files d'attente	15
1.2.6 Mesures de performance d'un système de file d'attente	16
1.3 Files d'attente avec "Working Vacation"	17
1.3.1 Politique de "Working Vacation"	17
1.3.2 Quelques modèles d'attente avec "Working vacation"	18
2 Optimisation numérique sans contraintes	23
2.1 Optimisation sans contraintes	23
2.1.1 Algorithmes de descente	25
2.1.2 Méthode de gradient	27
2.1.3 Méthode de gradient conjugué	28
2.1.4 Méthode de Newton	28
3 Optimisation de la file d'attente GI/M/1/N avec "working vacation"	35
3.1 Description de modèle	35
3.1.1 Description mathématique	35

3.2	Évaluation numérique des performances de la file GI/M/1/N avec "working vacation"	38
3.2.1	Implémentation numérique	39
3.3	Description et position du problème d'optimisation	46
3.3.1	Recherche de minimum	47
	Conclusion générale	51

Table des figures

1.1	Représentation schématique d'une file d'attente classique	12
1.2	Relation du coût total et du nombre des serveurs	15
2.1	Interprétation géométrique de la méthode de Newton	30
3.1	Représentation graphique de temps moyen de séjour GI/M/1/N avec "working vacation".	40
3.2	Représentation graphique de nombre moyen de clients dans la file d'attente GI/M/1/N avec "working vacation".	41
3.3	Représentation graphique de probabilité de blocage de modèle GI/M/1/N avec "working vacation".	42
3.4	Représentation graphique de probabilité de blocage dans la file d'attente GI/M/1/N avec "working vacation" en fonction de η	44
3.5	Représentation graphique de nombre moyen de clients dans la file d'attente GI/M/1/N avec "working vacation" en fonction de η	45
3.6	Représentation graphique de temps moyen de séjour dans la file d'attente GI/M/1/N avec "working vacation" en fonction de η	46
3.7	Représentation graphique de la fonction du coût.	50

Introduction générale

La théorie des files d'attente est l'un des outils analytiques les plus puissants pour la modélisation des systèmes dynamiques. Celle-ci a pour objet l'étude de systèmes et réseaux, où il y a des entités appelées clients cherchant à accéder à des ressources, généralement limitées, afin d'obtenir un service.

Ce domaine de recherche est apparu dans les années 1917, durant les travaux de l'ingénieur Danois Erlang sur la gestion des réseaux téléphoniques de Copenhague. C'est grâce aux apports des mathématiciens Khintchine, Palm, Kendall, Pollaczek et Kolmogorov que la théorie connaît vraiment un développement. La théorie des files d'attente est aujourd'hui largement utilisée et ses applications sont multiples.

Ces dernières années de nouvelles politiques sont ajoutées à l'étude des phénomènes d'attente. Parmi ces politiques, on trouve le système d'attente avec vacance ; ce genre de files d'attente ont une grande importance, parce qu'elles sont applicables dans l'analyse et la modélisation des systèmes informatiques, des réseaux de transmission, de la fabrication et des systèmes de production, des systèmes de transport, etc.

Les modèles de files d'attente avec vacances peuvent être classés en fonction des processus d'arrivée, des processus de service et les politiques de vacances. Des excellentes études sur les premiers modèles avec vacances ont été rapportées par Doshi [8], Takagi [37], Tian et Zhang [40]. Dans un système de files d'attente avec vacances, le serveur peut ne pas être disponible pendant une période de temps pour de nombreuses raisons comme, en cours de vérification pour la maintenance), on en travaillant dans d'autres files d'attente (c'est le cas de nombreux systèmes de communication), ou simplement prendre une pause dans cette période de temps. Lorsque le serveur est indisponible pour les clients ordinaires, le serveur est en vacances.

Dans l'étude du modèle d'attente avec vacances, on suppose généralement que le serveur arrête complètement le service au cours de la période des vacances. Cependant, il y

a de nombreuses situations, où le serveur ne restera pas complètement inactif au cours de cette période, mais il rendra le service à une file d'attente avec un taux de service différent ; c'est la politique des vacances : "working vacations" qu'on considérera dans ce mémoire. Cette politique a une grande importance, plus que celle des vacances normales, car les clients qui arrivent dans cette période seront servis avec un taux faible de service et non pas à quitter la file sans être servie, donc il seront satisfaits .

Les files d'attente avec "working vacation" ont été largement utilisés pour modéliser de nombreux problèmes pratiques, tels que les systèmes informatique, les systèmes de production, les systèmes de communication et d'autre domaine [34].

Dans ce travail, nous nous sommes intéressés à l'optimisation dans une file d'attente GI/M/1/N avec "working vacation", tout en utilisant une méthode d'optimisation classiques. Dans ce travail, nous allons utiliser les dérivées de la distribution stationnaire des chaînes de Markov à temps discret et à espace d'états fini [30]. Celle ci seront utilisés dans l'algorithme itératif de Newton-Raphson pour l'optimisation de la file d'attente GI/M/1/N avec "working vacation".

Ce mémoire est structuré de la manière suivante :

Le premier chapitre traite les notions de base de la théorie des files d'attente et la politique de "working-vacation".

Le deuxième chapitre est consacré à l'optimisation sans contrainte, et quelques méthodes d'optimisation, qu'on peut appliquer pour optimiser notre modèle.

Le troisième chapitre sera consacré à la présentation des résultats numériques concernant l'optimisation d'une fonction de coût associée au modèle d'attente GI/M/1/N avec "working vacation".

Systemes de file d'attente

La théorie des files d'attente est une théorie mathématique relevant du domaine des probabilités, qui étudie les solutions optimales de gestion des phénomènes d'attente. D'ailleurs, la théorie mathématique des files d'attente peut s'appliquer aux différentes situations : optimisation des stocks (gestion à flux tendu), gestion des avions , attente des clients à un guichet, ou bien encore traitement de données informatiques par un serveur. De plus, la modélisation des phénomènes d'attente est très importante dans la vie réelle, car elle permet d'organiser les systèmes ayant la structure d'une file d'attente.

Dans ce chapitre, nous allons présenter les éléments essentiels et quelques concepts et caractéristiques concernant les systèmes de files d'attente. Notre attention sera intéressée sur la présentation d'une bibliographie concernant les systèmes de files d'attente avec la politique de "Working vacation".

La modélisation des systèmes de files d'attente est basée sur les chaînes de Markov, donc on va rappeler d'abord les concepts et les propriétés principales des chaînes de Markov que nous avons besoin pour décrire une file d'attente

1.1 Chaîne de Markov finie

Définition 1.1.1 (processus stochastique) *Un processus stochastique (ou processus aléatoire) est une séquence X_1, X_2, \dots, X_N de variables aléatoires à valeur dans un espace dénombrable \mathbb{E} , L'ensemble \mathbb{E} est l'espace d'état, dont les éléments seront notés i, j, k, \dots Lorsque $X_n = i$, le processus est dit : être dans, ou visite l'état i au temps n .*

- Les valeurs possible des variables aléatoire sont appelées les états possibles du processus.

- La variable X_t représente l'état du processus au temps t (on dit aussi l'observation au temps t).
- L'intérêt des processus stochastiques est d'établir la dépendance entre les variables aléatoires.

Définition 1.1.2 (Chaîne de Markov) Une chaîne de Markov est un type particulier de processus stochastique qui vérifie deux conditions suivantes :

1. L'état au temps t du processus ne dépend que de son état au temps $t - 1$:

$$P(X_t = j | X_1 = i_1, \dots, X_{t-1} = i_{t-1}) = P(X_t = j | X_{t-1} = i_{t-1}).$$

2. La probabilité de passage d'un état i à un état j ne varie pas avec le temps

$$\forall t, 1 < t \leq N, \quad P(X_t = j | X_{t-1} = i) = c.c \in [0, 1]$$

On distingue deux types de chaîne de Markov :

- ✓ Une chaîne de Markov à temps discret lorsque t est discret.
- ✓ Une chaîne de Markov à temps continu lorsque t est continu.

On s'intéresse aux chaînes de Markov à temps discret qu'on peut définir comme suit :

Le processus stochastique $(X_n)_{n \geq 0}$ à valeur dans l'ensemble discret des états \mathbb{E} est une chaîne de Markov à temps discret \mathbb{N} si : $\forall n \in \mathbb{N}, \quad \forall (i_0, \dots, i_{n-1}, i, j) \in \mathbb{E}^{n+2}$,

$$P(X_{n+1} = j | X_n = i, X_{n-1} = i_{n-1}, \dots, X_0 = i_0) = P(X_{n+1} = j | X_n = i).$$

Définition 1.1.3 (La matrice de transition) Une matrice $P = p_{(ij)}_{i,j \in \mathbb{E}}$ où :

$$p_{ij} = P(X_{n+1} = j | X_n = i)$$

est la probabilité de transition de i vers j , est appelée matrice de transition de la chaîne de Markov. Les matrices de transition sont également appelées "matrices stochastiques" et satisfont les deux conditions suivantes :

1. Leurs éléments sont non négatifs :

$$p_{ij} \geq 0 \quad \forall i, j \in \mathbb{E};$$

2. La somme des éléments de chaque ligne est égale à 1

$$\sum_{j \in \mathbb{E}} p_{ij} = 1 \quad \forall i \in \mathbb{E}.$$

Définition 1.1.4 Deux états i et j d'une chaîne de Markov est dit communiquant, s'ils existent deux entiers m, n tel que :

$$p_{ij}^{(m)} = p_{ji}^{(n)}.$$

En d'autres termes, i communique avec j (on écrit $i \leftrightarrow j$), si on peut atteindre l'état j à partir de l'état i , et vice versa.

Définition 1.1.5 Une chaîne de Markov est irréductible, si et seulement si, pour tout état i et j , il existe $m \geq 0$ tel que :

$$p_{ij}^{(m)} > 0.$$

Autrement dit, une chaîne de Markov est irréductible si a la une seule classe de communication.

Définition 1.1.6 Une distribution de probabilité π sur \mathbb{E} est dite stationnaire si elle satisfait

$$\pi_j = \sum_{i \in \mathbb{E}} \pi_i p_{ij} \quad \forall j \in \mathbb{E}$$

tel que $\sum_{i \in \mathbb{E}} \pi_i = 1$.

Définition 1.1.7 (la matrice fondamentale d'une chaîne de Markov) cette matrice à été utilisée pour calculer les différentes caractéristiques stationnaires d'une chaîne de Markov d'espace d'états finis, la notion de matrice fondamentale a été introduite pour la première fois dans l'analyse des perturbations de chaîne de Markov par Snell et Kemeney [58]. Keilson [59] a élargi leur étude aux processus de Markov où il a obtenu de différentes formules des premiers moments de certaines caractéristiques. A l'origine la matrice fondamentale Z_θ a été utilisée pour résoudre le système d'équations $(I - P)X = b$, où b est connu. Pour calculer les valeurs de X , on a besoin de l'inverse de $(I - P)$ ce qui est impossible car il n'existe pas ce qui est conduit à modifier cette matrice tout en ajoutant la matrice Π (est une matrice contient dans chaque ligne le même vecteur de distribution

stationnaire) ce qui induit que l'inverse $Z_\theta = (I - P + \Pi)^{-1}$.

Quelques propriétés de la matrice fondamentale :

Si P_θ est une matrice de transition d'une chaîne de Markov finie et ergodique, alors la matrice inverse $Z = (I - P_\theta + \Pi_\theta)^{-1}$ existe, et :

1. $P_\theta Z_\theta = Z_\theta P_\theta$
2. $Z_\theta e = e$
3. $\pi_\theta Z_\theta = \pi_\theta$
4. $(I - P_\theta)Z_\theta = I - \Pi_\theta$
5. $Z_\theta = I + \sum_{i=1}^{\infty} (P_\theta^i - \Pi_\theta)$

Le théorème suivant établit la $k^{ième}$ dérivée de la distribution par rapport à θ .

Théorème 1.1.1 [30] [les dérivées successive de la distribution stationnaire π_θ] supposons que toutes les composantes de la matrice de probabilités de transition P_θ sont de classe C^k par rapport θ où θ est point intérieur d'un certain intervalle Θ avec $\Theta \in \mathbb{R}$. Alors, la $k^{ième}$ dérivée de la distribution stationnaire π_θ , d'une chaîne de Markov ergodique à temps discret et à espace d'états fini, par rapport au paramètre θ est donnée par :

$$\frac{d^k}{d\theta^k} \pi_\theta = \pi_\theta^k = \sum_{m=0}^{k-1} C_K^n \pi_\theta^{(m)} P_\theta^{(k-m)} Z_\theta \quad (1.1)$$

où $P_\theta^{(k)}$:est la matrice des dérivées $k^{ième}$ de la matrice de transition P_θ

$\pi_\theta^{(k)}$:est le vecteur des dérivée $k^{ième}$ de vecteur de distribution stationnaire π_θ .

1.2 Description du phénomène d'attente

Un phénomène d'attente peut être décrit comme un système composé d'un certain nombre (fini ou non) de places d'attente d'un ou plusieurs serveurs et de clients arrivant à des instants aléatoires. Les clients attendent, se font servir selon des règles spécifiées et quittent le système. Ils peuvent être des appels téléphoniques, des machines, ...,etc. de même que les serveurs peuvent être une centrale téléphonique, ...,etc. Quand les serveurs

sont tous occupés, les clients doivent alors patienter dans un espace d'attente (s'il existe) jusqu'à ce qu'un serveur soit disponible.

L'identification des systèmes de files d'attente classiques se base principalement sur trois éléments : le processus stochastique décrivant l'arrivée des clients dans le système, le mécanisme de service (le nombre de serveurs et la loi probabiliste décrivant la durée des services) et la discipline d'attente.

✓ Le processus d'arrivée spécifie les instants aux quels les clients arrivent dans le système. Dans la théorie classique des files d'attente, on fait le plus souvent l'hypothèse que les clients arrivent de manière isolée et indépendamment les uns des autres. Sous ces hypothèses, les intervalles de temps entre deux arrivées successives forment une suite de variables aléatoires indépendantes et identiquement distribuées.

✓ Les temps de service c'est les périodes nécessaire au traitement des clients qui sont supposés être des réalisations de variables aléatoires indépendantes et identiquement distribuées. La description du processus de service revient alors à préciser la loi de probabilité de ces variables aléatoires.

✓ Le nombre de serveurs correspond au nombre maximal de clients pouvant être traités simultanément. Tous les serveurs sont supposés identiques, en particulier les temps de service sont indépendants d'un serveur à l'autre et distribués selon une même loi de probabilité.

✓ La capacité d'accueil d'un système de files d'attente correspond au nombre maximal de clients pouvant être présents dans le système à un instant quelconque. Il est égal à la somme du nombre de serveurs et du nombre de places d'attente disponibles. Si un client arrive dans un système ayant atteint sa capacité maximale d'accueil, il est refoulé et doit quitter le système sans avoir été servi.

Quelques systèmes de file d'attente

- Gestion des avions et des trains au départ et à l'arrivée.
- Centre d'appels.
- Traitement des données dans les systèmes informatiques.

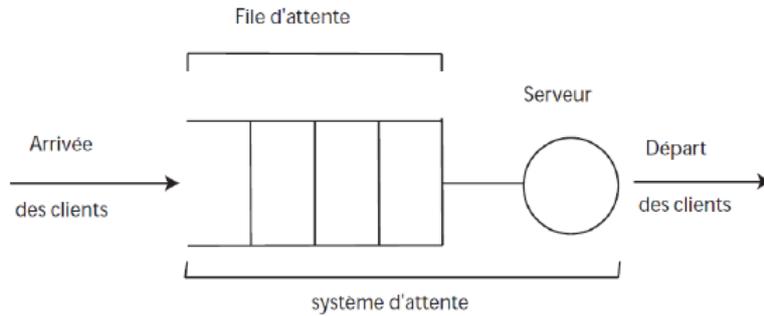


FIGURE 1.1 – Représentation schématique d'une file d'attente classique

- Attente pour un guichet ou une caisse.
- Commandes une attente de traitement.
- Patients dans une salle d'urgence.

1.2.1 Types de files d'attente

À partir de regroupement d'individus attendant un service, on pourra obtenir plusieurs types de files d'attente telles que :

File séparée

Une file par guichet (par exemple : caisses des supermarchés). Ce système a l'inconvénient de générer des frustrations lorsque certaines files sont plus rapides que d'autres, ou lorsqu'un guichet supplémentaire s'ouvre, permettant aux derniers de passer les premiers.

File distribuée (mutualisée)

Une seule file alimente plusieurs guichets.

File virtuelle

Une prise de ticket permet de conserver l'ordre d'arrivée, sans avoir à faire la file physiquement : par exemple, les personnes peuvent s'asseoir en attendant leur tour .

File virtuelle mobile

Les nouvelles technologies permettent maintenant de prendre rang par internet ou par téléphone, et d'être prévenu par SMS lorsque son tour approche. Le temps d'attente ne nécessite pas une présence physique .

File prioritaire

Des files plus rapides peuvent être créées, par exemple pour les personnes ayant un handicap, ou pour les personnes ayant une carte de fidélité ; parfois, des files prioritaires payantes peuvent être proposées.

1.2.2 Disciplines de service

Dans un système de files d'attente, une fois que le serveur devient libre, il choisit un client de la file suivant une politique ou discipline adoptée au sein du système . Les principales disciplines de service utilisées sont les suivantes :

- FIFO (First In First Out) : Premier Arrivé, Premier Servi, i.e., les clients sont servis dans leur ordre d'arrivée ;
- LIFO (Last In First Out) : Dernier Arrivé, Premier Servi, i.e., le dernier client arrivé sera le premier servi .
- Round-Robin(cyclique) :(RR) tous les clients de la file d'attente entrent en service à tour de rôle, effectuant un quantum "Q" de leur temps de service et ils sont remplacés dans la file, jusqu'à ce que leur service soit totalement accompli.
- PS (Processus Sharing) : cas limite de la discipline RR lorsque le quantum tend vers zéro. vitesse inversement proportionnelle au nombre de clients simultanément présents ;
- FIRO (First In Random Out) : Aléatoire, i.e., le prochain client qui sera servi est choisi aléatoirement dans la file d'attente.

1.2.3 Notation de Kendall

Pour la classification des types de systèmes de files d'attente, on utilise habituellement la notation de Kendall. Cette dernière est définie par la notation suivante :

$$A/B/C/(K/L/D_s)$$

Avec

- A : distribution d'inter-arrivée.
- B : distribution de service.
- C : nombre de serveurs.
- K : capacité de la file.
- L : population des usagers.
- D_s : discipline de service.

Où A et B sont donnés par

- M : loi exponentielle(markovienne)
- G : loi générale
- GI : lois générales indépendantes
- D : loi constante(déterministe)
- E_k : loi de erlang-k
- H_k : loi hyper-exponentielle-k
- C_k : loi de cox-k
- PH_k : loi de type phase à k étages

Remarque 1.2.1 Lorsque les trois derniers éléments de la notation de Kendall ne sont pas précisés, il est sous-entendu que $K = +\infty$, $m = +\infty$ et $Z = FIFO$.

1.2.4 Analyse économique de la théorie des files d'attente

L'objectif de la théorie des files d'attente est de satisfaire aux mieux les clients tout en minimisant ses coûts. Les coûts dans les modèles de la théorie d'attente se subdivisent en deux types :

Les coûts de service : c'est les coûts résultants du fonctionnement des serveurs des matériaux, appareils, des employés,etc, et c'est évident que ces coûts augmentent avec

l'augmentation du niveau de service.

Les coûts d'attente : ce sont les coûts résultants de la diminution de la qualité du service dans le système pour le servir, et elle contient la perte des clients qui quittent la file d'attente pour sa longueur et la perte d'autres clients qui les influencent et leur reportent la mauvaise réputation du système. La relation du coût total et du nombre des serveurs peut être représenté par le graphe suivant :

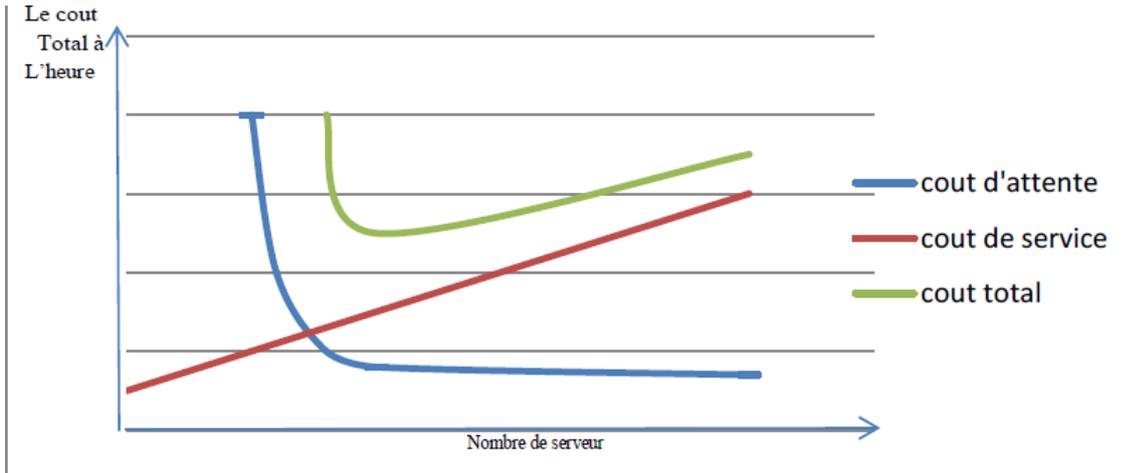


FIGURE 1.2 – Relation du coût total et du nombre des serveurs

1.2.5 Analyse mathématique d'un système de files d'attente

L'étude mathématique d'un système de files d'attente se fait généralement par l'introduction d'un processus stochastique, défini d'une façon appropriée. Généralement, on s'intéresse principalement au nombre de clients $N(t)$ se trouvant dans le système à l'instant $t(t \geq 0)$, En fonction des quantités qui définissent le système, qu'on cherche à déterminer.

Le régime transitoire du processus stochastique $N(t), t \geq 0$ défini par les probabilités d'état

$$P_n(t) = P_r[N(t) = n]$$

Le régime stationnaire du processus stochastique $N(t), t \geq 0$ défini par les distributions stationnaires de ce processus :

$$\pi_n = \lim_{t \rightarrow \infty} P_n(t) = P_r(N = n) \quad n = (0, 1, 2, \dots)$$

1.2.6 Mesures de performance d'un système de file d'attente

L'analyse théorique d'un modèle de file d'attente a pour objet de saisir qualitativement et quantitativement le fonctionnement du système en question. Pour cela, il faut définir les critères et les mesures susceptibles d'atteindre cet objectif afin de déterminer à l'avance les performances de système, pour un système composée d'une seule file d'attente, les principales mesures de performance sont :

- L : le nombre moyen de clients dans le système ;
- L_q : le nombre moyen de clients dans la file d'attente ;
- W : la durée moyen de séjour dans le système (attente et service) ;
- W_q : la durée moyen d'attente d'un seul client ;
- U : le taux d'occupation des serveurs ;
- P_{loss} : la probabilité de blocage pour les file d'attente à capacité finie ;
- P_{empty} : la probabilité que le système soit vide .

Une mesure très importante qui décrit le comportement asymptotique (lorsque il existe) du système tout entier, est le vecteur de la distribution stationnaire, noté par π .

Remarque 1.2.2 *La plupart des performance sont exprimées à l'aide de vecteur π .*

Formule de Little

Les valeurs caractéristiques d'un système de file d'attente sont liées les unes aux autres par les relations suivantes :

$$L = \lambda W,$$

$$L_q = \lambda W_q,$$

où λ est le taux d'arrivée de clients dans le système.

Ces deux formules sont appelées "Formules de Little". Ainsi, nous obtenons :

$$W = W_q + \frac{1}{\mu}.$$

où μ est le taux de service.

1.3 Files d'attente avec "Working Vacation"

Dans les files d'attente classiques le serveur est supposé qu'il a toujours un même rythme de travail c.-à-d. le taux moyen de service est le même pour toute les durées de service. D'autre part, le modèle en question a un taux moyen de service change dans certaines périodes de temps, c'est ces périodes de changement qu'on appelle "working vacation" notée WV.

1.3.1 Politique de "Working Vacation"

"Working-Vacation" est une nouvelle politique de vacances introduite en 2002 selon lequel le serveur fournit le service à une vitesse plus faible pendant la période de vacances plutôt que d'arrêter complètement le service. Ce type de modèles de vacances peut être appliqué dans l'évaluation des performances des réseaux de communication optique, systèmes de service, systèmes de production, systèmes informatiques tels que le service réseau, le service Web, le service de transfert de fichiers et le service de messagerie, et d'autres domaines. Ces dernières années, la recherche sur les files d'attente avec "Working Vacation" a connue des développements considérables. plusieurs modèle ont été étudiier dans cette zone.

De plus, cela démontre que l'approche analytique matricielle développée par Tian et al[14], est un outil puissant dans l'analyse des files d'attente avec "Working Vacation". En utilisant cette méthode, les distributions stationnaires et les structures de décomposition stochastique des principales mesures de performance peuvent être dérivées. Sur la base de ces résultats, un cadre théorique pour l'analyse des files d'attente de "Working Vacation" est établi.

Cette politique a été introduite par Servi et finn[34] dans un système de file d'attente $M/M/1$. Ils ont étudié une file $M/M/1$ avec "Working Vaction" et obtenu les formules de transformation pour la distribution du nombre de clients dans le système et le temps de séjour en régime stationnaire, et appliqué ces résultats à l'analyse de performance du routeur passerelle dans les réseaux de communication fibre. Pendant les périodes de "Working vacation", le serveur ne peut pas revenir à la période d'occupation normale avant la fin de la période de vacances. En outre, pendant la période WV, s'il y a des clients à un instant de fin de service, le serveur peut arrêter les vacances et revenir à l'état habituel.

Cette politique est appelée interruption de vacances.

1.3.2 Quelques modèles d'attente avec "Working vacation"

La politique de "working vacation" a été appliquée dans les différents modèles de file d'attente avec un seul serveur. Deux types de "Working Vacation" ont été utilisés : plusieurs "working vacation" (en anglais multiple working vacation que on note MWV) et un seul "working vacation" (en anglais single "working vacation" que on note SWV)

Modèles $M/M/1$ et $M/G/1$ avec Working Vacations

L'un des premiers modèles de file d'attente qui ont appliqué la politique de "Working Vacation" est le modèle $M/M/1$ avec multiple "working vacation". Ce modèle a été étudié pour la première fois par Servi et Finn[34] où ils ont obtenu la distribution de nombre moyen de clients dans le système stationnaire et la distribution du temps d'attente. Wu et Takagi[47] ont généralisé la file d'attente de Servi et Finn[34] $M/M/1/WV$ à une file d'attente $M/G/1/WV$. Kim et al[16] ont analysé la distribution de la longueur de file d'attente d'une file d'attente $M/G/1$ avec WV.

Liu et al.[28] ont dérivé les résultats de la décomposition stochastique dans une file d'attente $M/M/1$ avec WV. Par contre, Zhang et Xu [50] ont étudié une file d'attente $M/M/1$ avec plusieurs "working vacation" et la N politiques en utilisant un processus de quasi-naissance et de mort (QBD) et une méthode de solution géométrique matricielle. Une file d'attente $M/G/1$ avec exponentielle WV à été analysée par Li et al.[22] Jain et Jain[?] ont étudié une file d'attente avec "Working Vacation" et avec plusieurs types de pannes de serveur, où chaque type de panne nécessite un nombre aléatoire fini d'étapes de réparation. Pour les files d'attente d'arrivée par groupes. Xu et al[48] ont étudié une file d'attente $M^X/M/1$ avec single "working vacation" ils ont utilisé une méthode d'analyse matricielle. Baba [4] a prolongé la file d'attente d'arrivée de groupes $M^X/M/1$ avec single working vacation à une file d'attente $M^X/M/1$ avec multiple working vacation. Récemment, Gao et Yao[11] ont développé une file d'attente $M^X/G/1$ avec "working vacation" aléatoires.

Pour les modèles d'interruption de vacances, Li et Tian[20].l'ont introduit pour la première fois dans une file d'attente $M/M/1$ c.à-d. Pendant une période WV, le serveur peut interrompre les vacances à l'instant de fin de service s'il y a des clients dans le

système ,c'est ce qu'on appelle "Working-Vacation" avec interruption de vacances, cette dernière politique a été appliquée à une file d'attente $M/G/1/MWV$ par Zhang et Hou .[51] Tian et al [41] ont étudié une file d'attente $M/M/1$ avec SWV en utilisant une méthode géométrique matricielle.Zhang et Hou[54] ont analysé une file d'attente $M/G/1$ avec "working-vacation" et interruption de vacances. En utilisant la méthode d'une variable supplémentaire et la méthode analyse matricielle,le nombre moyen de clients dans le système et l'état de service à l'état stationnaire sont obtenus.

Li et al[24], ont étudié une file d'attente $M/G/1$ avec un temps de préparation et de multiples working-vacation. Ils ont obtenu le nombre de clients dans le système aux instants de départ, puis ils ont déduit la matrice de transition d'état en utilisant la méthode de la chaîne de Markov induite. Gao et Liu[9] ont récemment introduit une file d'attente $M/G/1$ avec une seul working vacation et avec une politique d'interruption de vacance de Bernoulli. Sur la base des travaux précédents, Lee et Kim [17] ont étudié la distribution du temps de séjour d'une file d'attente $M/G/1$ avec une seul working vacation et une interruption de vacances, ils ont dérivé le LST(Transformer de Stieltjes-Laplace) explicite du temps de séjour, distribution de ce modèle.Selvaraju et Cosmika [33] ont analysé une file d'attente de clients impatientes $M/M/1$ avec working vacation simples et multiples.

Yang et Wu[49] ont étudié l'analyse de minimisation des coûts de la file d'attente $M/M/1/WV$ et N politiques avec rupture de serveur. Zhang et Liu [55] ont développé "working vacation" et l'interruption de vacances dans la file d'attente $M/G/1$ avec des clients négatifs et la panne du serveur.

Les files d'attente $GI/M/1$ et $GI/G/1$ avec "working vacation"

Pour les modèles d'entrée générale, Baba[3] a étudié une file d'attente $GI/M/1$ avec "working vacation" par la méthode d'analyse matricielle. Banik et al [5] ont analysé de la file d'attente $GI/M/1/N$ avec "working vacation" en présentant une série de résultats numériques. Li et Tian[19] ont établi un lien entre "working vacation" et interruption de vacances et ont analysé la file d'attente $GI/Geom/1$ en temps discret avec "working vacation" et interruption de vacances. Plus tard, Li et al[21]. ont analysé la file d'attente $GI/M/1$ avec "working vacation" et l'interruption de vacances. En utilisant la méthode d'analyse matricielle, ils ont obtenu diverses mesures de performance telles que le nombre moyenne de clients dans le système et le temps moyen d'attente. En outre, la période

où il commence le "working vacation" pour la file d'attente $GI/M/1$ avec interruption de vacances et "working vacation" proposée par Zhao et al .[56] Chen et al[6] ont exprimé la file $GI/M/1$ avec PH "working vacation" et interruption de vacances telle que le temps de vacances suit une distribution de type phase (PH) . Li et Tian [23] ont discuté la file d'attente $GI/M/1$ avec "working vacation" . En utilisant l'approche analytique matricielle, les distributions en régime stationnaire est obtenu de nombre de clients dans le système à l'arrivée et à l'époque arbitraire .

Zhang et Hou [52] ont étudié une file d'attente $GI/M/1/N$ avec une variante de multiple "working vacation" , en utilisant la variable supplémentaire et les méthodes de chaîne de Markov induite, où ils ont obtenu la distribution de longueur de file d'attente à différentes époques.

Tao et al[38] ont proposé une nouvelle file d'attente $GI/M/1$ avec une période de démarrage et "working vacation" et une interruption de vacances en même temps. À l'aide de la méthode matricielle-analytique, ils ont obtenu les distributions en régime stationnaire pour le nombre moyen de clients dans la file d'attente, le temps d'attente et le temps de séjour.

Vijayalaxmi et Jyothisna[44] ont analysé la file d'attente à capacité finie $GI/M/1$ avec multiple "working vacations" qui a des applications potentielles dans la fabrication, le transport, les systèmes de communication, etc. Plus tard en 2013, Vi-jayalaxmi et al[45] ont mis en œuvre un modèle intégrant "working vacation" individuelle et l'interruption de vacances.

Récemment, Vijayalaxmi et Suchitra [46] ont étudié le modèle de la file d'attente $GI/M(n)/1$ avec multiple "working vacation" et temps de changement de service, où le serveur peut continuer à fonctionner mais à une vitesse moins pendant les vacances.

Files d'attente avec rappel(nouveau procès) avec "working vacation"

Des files d'attente de vacances et des files d'attente de nouveau procès ont été appliquées pour évaluer les performances de divers systèmes. Dans les études récentes, beaucoup de chercheurs ont discuté du concept de files d'attente de nouveau procès avec "working vacation".

Le système avec rappels est caractérisé par un client arrivant qui trouve le serveur occupé, quitte la zone de service et répète sa demande après un certain temps. Entre les

essais, le client bloqué rejoint un pool de travaux insatisfaits appelés orbite, par exemple, accès web, réseaux de télécommunication, réseaux de commutation de paquets, évitement de collision, réseaux locaux en étoile, etc.

La file d'attente $M/M/1$ avec rappel et "working vacation" a été introduite et analysée par Do[7]. Li et al[24] ont donné une file de réapprovisionnement $Geo/Geo/1$ avec rappel et "working vacation" et interruption de vacances.

Liu et Song [29] ont introduit des clients non persévérants et des "working vacation" dans la file d'attente de réapprovisionnement en temps discret $Geo/Geo/1$ avec rappel. En utilisant la méthode matricielle-analytique, ils ont obtenu la distribution de probabilité stationnaire et quelques mesures de performance et ont montré la décomposition stochastique conditionnelle pour la longueur de la file d'attente en orbite.

Récemment, Arivudainambi et al.[2] ont d'abord exploré un système de file d'attente $M/G/1$ avec rappel et individuelle "working vacation". Gao et al[10] ont pris en considération une file d'attente $M/G/1$ avec rappel avec des délais général de rappel, et introduit "working vacation" et l'interruption des vacances dans la liste des nouveaux procès (rappel). Utilisation de méthode de variable supplémentaire pour obtenir la distribution de probabilité stationnaire et certaines mesures de performance. De plus, ils ont trouvé la distribution du temps d'attente et ont prouvé la décomposition stochastique conditionnelle de nombre moyen de clients dans la file d'attente en orbite.

Aissani et al[1]. ont discuté une file d'attente $M/G/1$ avec rappel et "working vacation". Ils ont obtenu des fonctions génératrices permettant d'obtenir plusieurs autres mesures de performance telles que l'attente moyenne ou le temps moyen de séjour.

Très récemment, Rajadurai et al [31]et[32] ont mis au point des modèles de mise en attente des nouveaux procès (rappel) avec le concept de "working vacation" et d'interruption de vacances dans les cas de panne.

Autres modèles de "working vacation"

Un autre développement récent dans la recherche de modèle de "working vacation" est sur le système avec le processus d'arrivée de Markov (MAP). Zhao et Cui [57] ont étudié l'approximation du PH pour la file d'attente $MAP/G/1$ avec "working vacation" et N-politiques.

Zhang et Hou [53] ont étudié la file d'attente $MAP/G/1$ avec "working vacation"et

interruption de vacances. Ils ont obtenu la distribution de nombre moyen de clients dans la file d'attente avec la méthode de variable supplémentaire, combinée avec la méthode d'analyse matricielle et la technique de "censoring" et ont également trouvé la distribution de la taille du système à l'époque pré-arrivée et la transformation Laplace-Stieltjes (LST) du temps d'attente.

Sreenivasan et al [35] ont élargi le travail de Li et Tian[20] aux arrivées de MAP et aux services de type phase et ont introduit les vacances en N-politique dans ce modèle.

Récemment, Liu et al.[27] étudié un système réparable froid standby avec "working vacation" et interruption de vacances suivant un processus d'arrivée Markovien.

Conclusion

Dans le modèle "working vacation" la maintenance du serveur peut être effectuée sans l'insatisfaction de client, ce modèle réduit la possibilité de retour des clients par rapport au vacances normales. En effet en cette période de fortes demande pour les produits et services qui sont disponibles pour une petite durée de temps, le modèle avec "working vacation" est le plus utilisé. A cet effet, des vastes travaux de recherche qui ont été réalisés sur cette politique .

Optimisation numérique sans contraintes

2.1 Optimisation sans contraintes

L'objectif d'optimisation sans contraintes de fonctions continues non linéaires est de trouver un vecteur X^* tel que $f(X^*)$ soit un extremum de f (f est une fonction continue définie de \mathbb{R}^n vers \mathbb{R}), tous les algorithmes présentés ici cherchent un minimum de la fonction objectif. Autrement dit, on cherche X^* tel que :

$$X^* \in \arg \min_{X \in \mathbb{R}^n} f(X).$$

Trouver un minimum global est en général un problème très difficile. En réalité, il n'existe pas d'algorithme d'optimisation parfait, en fonction du problème, on cherchera donc la méthode la mieux adaptée. Ce chapitre traite des méthodes numériques (Algorithmes) permettant de trouver rapidement un minimum local, ces algorithmes sont couramment utilisés par des méta-heuristiques (kangourou, recuit simulé, etc.) afin de trouver le (le minimum ou les minimums locaux).

La présentation des algorithmes s'organise autour des critères suivants :

- les algorithmes utilisés sont différents selon que la fonction objectif est mono-variable ou multivariée,
- les algorithmes qui utilisent uniquement la valeur de la fonction sont dits d'ordre zéro,
- les algorithmes qui utilisent la valeur de la fonction et la valeur de sa dérivée (de son gradient dans \mathbb{R}^n) sont dits d'ordre un,

- les algorithmes qui utilisent la valeur de la fonction, de sa dérivée et de sa dérivée seconde (de son hessien dans \mathbb{R}^n) sont dits d'ordre deux.

Dans ce qui se suit on va étudier quelques méthodes très utilisées dans l'optimisation sans contraintes pour la recherche d'un minimum d'une fonction ; dans ce qui suit, nous allons donner quelques rappels sur des notions de base qu'on a besoin dans ces méthodes

Définition 2.1.1 (Gradient d'une fonction) *Si f est une fonction réelle dans \mathbb{R}^n , le gradient de f est la fonction vectorielle :*

$$\nabla f(X) = \begin{pmatrix} \frac{\partial f}{\partial x_1}(X) \\ \frac{\partial f}{\partial x_2}(X) \\ \vdots \\ \frac{\partial f}{\partial x_n}(X) \end{pmatrix}.$$

Définition 2.1.2 (Point critique) *Soit $f : \mathbb{R}^n \rightarrow \mathbb{R}$ une fonction différentiable. Tout point $X^* \in \mathbb{R}^n$ tel que $\nabla f(X) = 0$ est appelé point critique ou point stationnaire de f .*

Définition 2.1.3 (Hessien) *Soit $f : \mathbb{R}^n \rightarrow \mathbb{R}$, une fonction deux fois différentiable. La fonction notée $\nabla^2 f(X) : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times n}$ est appelée matrice hessienne ou hessien de f et est définie par :*

$$\nabla^2 f(X) = \begin{pmatrix} \frac{\partial^2 f}{\partial x_1^2}(X) & \frac{\partial^2 f}{\partial x_1 \partial x_2}(X) & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n}(X) \\ \frac{\partial^2 f}{\partial x_2 \partial x_1}(X) & \frac{\partial^2 f}{\partial x_2^2}(X) & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n}(X) \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1}(X) & \frac{\partial^2 f}{\partial x_n \partial x_2}(X) & \cdots & \frac{\partial^2 f}{\partial x_n^2}(X) \end{pmatrix}.$$

La matrice hessienne est symétrique par définition.

Définition 2.1.4 *La matrice hessienne est dite semi-défini positive si :*

$$\forall X \in \mathbb{R}^n : X^T \nabla^2 f(X) X \geq 0.$$

La matrice hessienne est dite défini positive si :

$$\forall X \in \mathbb{R}^n : X^T \nabla^2 f(X) X > 0$$

Théorème 2.1.1 [13][Conditions nécessaires d'optimalité d'ordre 1] Soit $f : \mathbb{R}^n \rightarrow \mathbb{R}$ différentiable en $X^* \in \mathbb{R}^n$, Si X^* est un minimum local de f alors :

$$\nabla f(X^*) = 0.$$

Théorème 2.1.2 [13] [Conditions nécessaires d'optimalité d'ordre 2] Soit $f : \mathbb{R}^n \rightarrow \mathbb{R}$ deux fois différentiable au point $X^* \in \mathbb{R}^n$, Si X^* est un minimum local de f alors : $\nabla f(X) = 0$ et $\nabla^2 f(X)$ est semi-défini positive.

Théorème 2.1.3 [13] [Conditions suffisantes de minimalité locale] soit $f : \mathbb{R}^n \rightarrow \mathbb{R}$ une application deux fois différentiable, Si $X^* \in \mathbb{R}^n$ vérifie les conditions :

$$\begin{cases} \nabla f(X^*) = 0, \\ \nabla^2[f](X^*) \quad \text{est définie positive} \end{cases}$$

Alors, X^* est un minimum local strict de f .

Remarque 2.1.1 Pour les fonctions f non convexe, il faut vérifier les deux conditions suffisantes de minimalité locale pour dire que X^* est un minimum local. Dans le cas où la fonction f est convexe il faut vérifier seulement si $\nabla f(X^*) = 0$, pour dire que X^* est un minimum local, et si f est convexe, alors tout minimum local est aussi global. De plus, si f est strictement convexe, alors tout minimum local devient minimum global unique.

Remarque 2.1.2 Dans le cas où f est une fonction mono-variable (f définie de \mathbb{R} vers \mathbb{R}) on remplace les conditions par la première dérivée et la deuxième dérivée.

Remarque 2.1.3 D'un point de vue géométrique, la condition du second ordre : " $\nabla^2[f](X^*)$ définie positive", revient à dire que f est localement convexe en X^* i.e. convexe dans un voisinage ouvert de X^* . En pratique, il est difficile de vérifier systématiquement car il nécessite de calculer les dérivées secondes et d'étudier les valeurs propres de la matrice hessienne.

2.1.1 Algorithmes de descente

Ce sont des algorithmes utilisés pour calculer les minimums d'une fonction, un algorithme de descente part d'un point de départ X_0 choisi arbitrairement et cherche à générer

une suite d'itérations $(X_k)_{k \in \mathbb{N}}$ telle que :

$$X_{k+1} = X_k + \alpha_k d_k, \text{ où } \alpha_k \geq 0,$$

tout en assurant la propriété $f(X_{k+1}) < f(X_k)$.

Le vecteur d_k est la direction de descente en X_k , le scalaire α_k est appelé le pas de la méthode à l'itération k .

Définition 2.1.5 (direction de descente) *Soit $f : \mathbb{R}^n \rightarrow \mathbb{R}$ une fonction différentiable.*

Soit $X, d \in \mathbb{R}^n$. La direction d est une direction de descente en X si :

$$d^T \nabla f(X) < 0.$$

La direction du gradient est celle dans laquelle la fonction a la plus forte pente. La direction opposée au gradient est donc celle dans laquelle la fonction a la plus forte descente.

Théorème 2.1.4 [13][descente]

Soit $f : \mathbb{R}^n \rightarrow \mathbb{R}$ une fonction différentiable. Soient $X \in \mathbb{R}^n$ tel que $f(X) \neq 0$ et $d \in \mathbb{R}^n$. Si d est une direction de descente en X , alors il existe η tel que :

$$\forall \alpha \in [0, \eta], f(x + \alpha d) < f(x).$$

En utilisant ce résultat, on construit un algorithme général de minimisation, nommée algorithme de descente, il consiste simplement à suivre une direction de descente de façon itérative jusqu'à l'obtention d'une bonne minimisation. Habituellement on utilise plusieurs critères d'arrêt. on trouvera :

- un critère sur le déplacement, si $\|X_{k+1} - X_k\|$ est très petit, c'est qu'on ne progresse plus beaucoup ;
- un critère sur la progression de l'objectif, si $|f(X_{k+1}) - f(X_k)|$ est très petit, on peut être presque arrivé à un minimum. On peut aussi mesurer la norme du gradient ;
- un critère sur le temps de calcul ou le nombre d'itérations.

Algorithm 1 Algorithme de descente

f une fonction différentiable et X_0 un point initial .
 $k \leftarrow 0$;
tant que le critère d'arrêt non satisfait **faire**
 Trouver une direction de descente d_k telle que $d_k^T \nabla f(X_k) < 0$;
 Trouver un pas α_k tel que : $f(X_k + \alpha_k d_k) < f(X_k)$;
 $X_{k+1} \leftarrow X_k + \alpha_k d_k$;
 $k \leftarrow k + 1$;
fin.

2.1.2 Méthode de gradient

Supposons que la fonction $f : \mathbb{R}^n \rightarrow \mathbb{R}$ est différentiable. Un algorithme de type gradient utilise comme direction de descente l'opposée de la direction du gradient de f au point courant :

$$X_0 \in \mathbb{R}^n \quad \text{quelconque}, \quad X_{k+1} = X_k - \alpha_k \nabla f(X_k).$$

A chaque itération, le point d'arrêt devient le point de départ où la fonction est réévaluée et une nouvelle direction est suivie. Le processus est répété jusqu'à ce que le minimum soit atteint.

Schématiquement on peut résumer le processus ainsi :

Algorithm 2 Algorithme de gradient

f une fonction différentiable et X_0 un point initial.
 $X \leftarrow X_0$;
tant que le critère d'arrêt non satisfait **faire**
 $d \leftarrow -\nabla f(X)$;
 Trouver un pas α qui minimise $f(X + \alpha d)$;
 $X \leftarrow X + \alpha d$;
fin.

Il existe différentes méthodes de gradient se distinguent d'après le choix de la longueur de pas :

Méthode de gradient à pas fixe

Dans cette méthode, le pas de déplacement est le même pour toutes les itérations ie :

$$\alpha_k = \alpha \quad \forall k = 0, 1, 2, \dots$$

Les itérations sont donc construites à partir de la formule :

$$X_{k+1} = X_k - \alpha \nabla f(X_k), \quad k = 0, 1, 2, \dots$$

Méthode de gradient à pas optimal

L'idée de l'algorithme du gradient à pas optimal est d'essayer de calculer à chaque itération le paramètre qui minimise la fonction dans la direction de descente donnée par le gradient. Les itérations sont construites à partir de la relation :

$$X_{k+1} = X_k - \alpha_k \nabla f(X_k), \quad k = 0, 1, 2, \dots$$

où α_k est choisi de façon à minimiser la fonction f la long de la direction de descente

$$d_k = -\nabla f(X_k)$$

le pas α_k est solution du problème :

$$\min_{\alpha > 0} \phi(\alpha) = \min f(X_k - \alpha \nabla f(X_k)) = f(X_k - \alpha \nabla f(X_k))$$

L'inconvénient de cet algorithme est que la convergence peut être très lent(en générale est de type linéaire) pour certaines types de fonctions

Méthode de gradient à pas variable

La longueur de pas varie au cours des itérations, d'après certains critères.

2.1.3 Méthode de gradient conjugué

La méthode du gradient conjugué est une méthode de descente à pas optimal permettant de minimiser une fonction quadratique de \mathbb{R}^n dans \mathbb{R} en au plus n itération.

2.1.4 Méthode de Newton

La méthode de Newton n'est pas une méthode d'optimisation proprement parler. C'est en réalité une méthode utilisée pour résoudre des systèmes non linéaires de la forme

Algorithm 3 Algorithme du gradient conjuguée (Fletcher-Reeves)

f une fonction différentiable et X_0 un point initial ;

$X \leftarrow X_0$;

$G \leftarrow \nabla f(X)$;

$d \leftarrow -G$;

$s \leftarrow \|G\|^2$;

Tand que critère d'arrêt non satisfait **Faire**

Trouver un pas α qui minimise $f(X + \alpha d)$;

$X \leftarrow X + \alpha d$;

$G \leftarrow \nabla f(X)$;

$t \leftarrow s$;

$s \leftarrow \|G\|^2$;

$d \leftarrow -G + \frac{s}{t}d$;

résultat X .

$f(x) = 0$ où f est une fonction de \mathbb{R}^n dans \mathbb{R}^n . Nous décrivons d'abord la méthode de Newton pour trouver une racine d'une fonction à une variable, puis la méthode analogue pour la recherche d'un minimum, d'abord à une dimension, puis à n dimensions.

Méthode de Newton pour trouver une racine d'une fonction à une variable

Étant donné une fonction $f : \mathbb{R} \rightarrow \mathbb{R}$ et un point initiale x_0 proche de la solution. La fonction calcule la tangente au point (première dérivée de la fonction). La prochaine valeur à utiliser est l'intersection de la tangente et l'axe des x . C'est un processus itératif (itérations successives) qui va s'arrêter quand le critère de convergence soit atteint. La méthode de Newton peut être utilisée dans le cas d'une fonction continue et différentiable dans l'intervalle de recherche $[a, b]$

$$f : [a, b] \rightarrow \mathbb{R}.$$

La condition de différentiabilité assure la présence d'une droite tangente à notre fonction. La condition de continuité assure qu'il n'y a pas de sauts dans la fonction.

La tangente de la courbe (dérivée de la fonction) s'obtient :

$$f'(x_n) = \frac{\Delta y}{\Delta x} = \frac{f(x_n)}{x_n - x_{n+1}}$$

Qu'on peut écrire

$$y = f'(x_n)(x_{n+1} - x_n) + f(x_n).$$

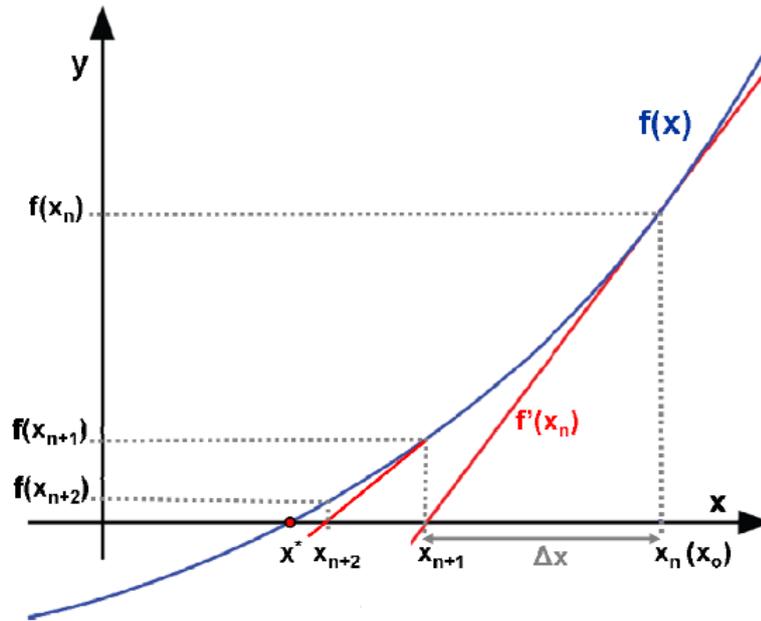


FIGURE 2.1 – Interprétation géométrique de la méthode de Newton

Cette tangente coupe l'axe des abscisse quand $Y = 0$.

$$f'(x_n)(x_{n+1} - x_n) + f(x_n) = 0 \Leftrightarrow f'(x_n)(x_{n+1} - x_n) = -f(x_n).$$

$$(x_{n+1} - x_n) = \frac{-f(x_n)}{f'(x_n)} \Leftrightarrow x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}.$$

On peut écrire la méthode de Newton comme

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \quad \text{où } n \geq 0. \quad (2.1)$$

L'intervalle de recherche $[a, b]$, peut être choisi en utilisant le théorème des valeurs intermédiaires qui énonce que pour une fonction f continue dans un intervalle $[a, b]$ si $f(a) * f(b) < 0$ (ou l'inverse), il existe au moins une valeur $c \in [a, b]$ qui confirme $f(c) = 0$. Dans les grandes lignes de cette méthode sont données dans l'algorithme suivant :

En général, cette procédure a une convergence très rapide (convergence d'ordre quadratique), surtout quand la valeur initiale est proche de la solution. Malgré ça, elle présente quelques difficultés :

- Pour les approximations initiales loin de la solution, la méthode peut présenter une

Algorithm 4 Algorithme de Newton

Choisir une approximation initiale x_0 .

Calculer la valeur de la fonction en ce point $f(x_0)$.

Calculer la dérivée de la fonction f et évaluer la valeur au point $f'(x_0)$.

L'approximation x_1 s'obtient avec $x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$.

Répéter la procédure avec $x_1, x_2, x_3 \dots$ jusqu'à que le critère d'arrêt est satisfait.

non convergence

- La dérivée de la fonction doit être calculée directement. Parfois, cette dérivée n'est pas exacte et on utilise alternativement des méthodes approximatives du type sécant (on prend la sécant au lieu de la tangent pour chaque itération) avec un ordre de convergence linéaire.
- Si la dérivée de la fonction n'est pas continue ou nulle, la méthode n'est pas utilisable.

Nous pouvons maintenant généraliser à \mathbb{R}^n , Soit f une fonction de classe \mathcal{C}^1 définie de \mathbb{R}^n à valeurs dans \mathbb{R}^n , On suppose que l'équation :

$$f(x) = 0$$

possède au moins une solution notée x^* et que la matrice jacobienne $Df(x^*)$ est une matrice inversible, la continuité de Df permet alors d'assurer l'inversibilité de $Df(x_k)$ pour tout x_k dans un voisinage de x^* .

Algorithm 5 Algorithme de Newton dans \mathbb{R}^n

Choisir une approximation initiale x_0 dans un voisinage de x^* .

Calculer la valeur de la fonction en ce point $f(x_0)$.

Calculer $Df(x)$ et évaluer les valeurs au point $Df(x_0)$.

L'approximation x_1 s'obtient avec $x_1 = x_0 - [Df(x_0)]^{-1}f(x_0)$.

Répéter la procédure avec $x_1, x_2, x_3 \dots$ jusqu'à que le critère d'arrêt est satisfait.

Méthode de Newton pour trouver un minimum d'une fonction à une variable

Étant donné une fonction $f : \mathbb{R} \rightarrow \mathbb{R}$ et un point initiale x_0 , nous appliquons la méthode de Newton, cette fois-ci pour trouver un minimum local de f . Puisque la condition suffisante d'un minimum local est que la dérivée première soit nulle, donc on va utiliser la méthode précédente pour chercher une racine de la dérivée de f . Remplacer f

par f' et f' par f'' dans l'équation donne alors :

$$x_{k+1} = x_k - \frac{f'(x_k)}{f''(x_k)}.$$

Une autre manière d'obtenir la même formule est d'utiliser la formule de Taylor à l'ordre 2 de f :

$$f(x_k + h) = f(x_k) + f'(x_k)h + f''(x_k)\frac{h^2}{2} + \epsilon.$$

Le but est alors de trouver h^* qui minimise $f(x_k) + f'(x_k)h + f''(x_k)\frac{h^2}{2}$. La dérivée par rapport à h de cette expression doit alors vérifier :

$$f'(x_k) + f''(x_k)h = 0$$

d'où le résultat.

$$h^* = -\frac{f'(x_k)}{f''(x_k)}.$$

Algorithme 6 Algorithme de Newton pour la recherche d'un minimum

f une fonction de classe \mathcal{C}^2 et x un point dans le voisinage d'un minimum de f .

Répéter

$y \leftarrow x$;

$x \leftarrow x - \frac{f'(x)}{f''(x)}$;

jusqu'à $|x - y| < \epsilon$;

Résultat : x .

D'un point de vue pratique, cette méthode souffre de nombreux inconvénients :

- la méthode peut diverger si le point de départ est trop éloigné de la solution,
- la méthode n'est pas définie si $f''(x_k) = 0$,
- la méthode peut converger indifféremment vers un minimum, un maximum ou un point selle.

Méthode de Newton dans \mathbb{R}^n

La méthode de Newton peut être généralisée au cas de fonctions f définie de \mathbb{R}^n vers \mathbb{R} . La méthode de Newton pour les fonctions multivariées est identique à celle des fonctions uni variées. Comme précédemment, une approximation de deuxième ordre de f au point

X_k peut s'écrire sous la forme :

$$f(X_k + h) = f(X_k) + h^T \nabla f(X_k) + \frac{1}{2} h^T [\nabla^2 f(X_k)] h + \epsilon.$$

Si la matrice hessienne est inversible, alors on trouve :

$$h^* = [\nabla^2 f(X_k)]^{-1} \nabla f(X_k),$$

d'où la formule de Newton qui minimise f

$$X_{k+1} = X_k - [\nabla^2 f(X_k)]^{-1} \nabla f(X_k).$$

Algorithm 7 Algorithme de Newton à n dimension

f une fonction différentiable et X_0 un point initial.

$X \leftarrow X_0$,

calculer le gradient $\nabla f(X)$,

calculer le hessien $\nabla^2 f(X)$,

calculer l'inverse de hessien

$X \leftarrow X + [\nabla^2 f(X)]^{-1} \nabla f(X)$,

fin

Résultat : X .

L'intérêt de cette suite est sa convergence quadratique vers un minimum local à condition que X_0 soit assez proche d'un minimum. Cette méthode comporte les mêmes inconvénients que dans le cas mono variable :

- ✓ la méthode peut diverger si le point de départ est trop éloigné de la solution,
- ✓ la méthode n'est pas définie si la matrice hessienne n'est pas inversible,
- ✓ la méthode peut converger indifféremment vers un minimum, un maximum ou un point selle.

Remarque 2.1.4 *La méthode de Newton est un algorithme de descente à pas fixe égal à 1.*

Si la fonction f est une fonction quadratique est strictement convexe, alors l'algorithme converge en une seule itération.

Conclusion

C'est très important de visualiser la fonction que l'on veut optimiser avant de choisir l'un des algorithmes et pour déterminer la valeur initiale à introduire pour notre analyse. D'ailleurs le même algorithme peut s'utiliser pour résoudre différents types de problèmes, optimisation, calcul de racines, solutions d'un système d'équations...

Optimisation de la file d'attente

GI/M/1/N avec "working vacation"

Dans cette partie numérique de ce manuscrit, nous utilisons la méthode de Newton, pour illustrer l'utilité des dérivées de la distribution stationnaire de système d'attente GI/M/1/N avec "working vacation".

En effet L'objectif de notre travail est de chercher à optimiser une fonction coût dans un modèle d'attente GI/M/1/N avec "working vaction"par rapport au taux moyen de service pendant "working vacation".

3.1 Description de modèle

Considérons une file d'attente GI/M/1/N, où le système est décrit par : une capacité limité de la file notée N , un temps des inter-arrivées qui ce font selon une loi générales noté G et un temps de service Markovien qui peut avoir deux cas :

1^{ier} cas : Le serveur est en période de service normal

Si le système est non vide, le serveur servira les clients présents dans le système avec un taux moyen μ

$2^{ième}$ cas : Le serveur est en période de "working vacation" :

Dans ce cas le serveur fait un taux moyen de service faible noté η . En plus si le serveur trouve le système vide après son retour il continuera la période de "working vacation".

3.1.1 Description mathématique

On note

λ : taux moyen des inter-arrivées,

μ : taux moyen de service pendant la période de service normal,

η : taux moyen de service pendant "working vacation",

γ : paramètre qui mesure le temps de "working vacation",

Remarque 3.1.1 *Les temps des inter-arrivées $T_{n+1} = t_{n+1} - t_n$, où $n = 0, 1, 2, \dots$ sont des variables aléatoires indépendantes identiquement distribuées selon la même loi qu'on note $A(t)$.*

Remarque 3.1.2 *Le temps de service pendant la période de service, temps de service pendant "working-vacation", et les temps de "working-vacation" sont supposés être distribués de façon exponentielle.*

Les états de ce système à l'instant t est décrit par la chaîne de Markov $X_t = (X_s(t), \epsilon(t))$, où :

$$\epsilon(t) = \begin{cases} 1 & \text{si le serveur est en période de service normal;} \\ 0 & \text{si le serveur est en "working-vacation".} \end{cases}$$

$X_s(t)$:représente le nombre de clients dans la file plus le client qui est en service.

$(X_s(t), \epsilon(t))$ forme une chaîne de Markov d'espace d'états finis, définies comme suite :

$$((0, 0), (1, 0), (1, 1), (2, 0), (2, 1), \dots, (N, 0), (N, 1)).$$

Dans ce cas définissons

$$\pi_{(N,1)} = \lim_{t_i \rightarrow \infty} P(N_s(t_i) = n, \epsilon(t_i) = 1), \quad 1 \leq n \leq N;$$

$$\pi_{(N,0)} = \lim_{t_i \rightarrow \infty} P(N_s(t_i) = n, \epsilon(t_i) = 0), \quad 0 \leq n \leq N;$$

La distribution stationnaire π de la chaîne de Markov décrivant l'état du système étudié est donnée sous la forme suivante :

$$\pi = (\pi_{(0,0)}, (\pi_{(1,0)}, \pi_{(1,1)}), (\pi_{(2,0)}, \pi_{(2,1)}), \dots, (\pi_{(N,0)}, \pi_{(N,1)})$$

. Soient a_k, b_k, c_k les probabilité conditionnelles définies comme suit :

$$a_k = \int_0^{+\infty} \frac{(\mu t)^k}{k!} e^{-\mu t} dA(t).$$

$$b_k = \int_0^{+\infty} e^{\gamma t} \frac{(\eta t)^k}{k!} e^{-\eta t} dA(t).$$

$$c_k = \int_0^{+\infty} \sum_{j=0}^k \int_0^t \gamma e^{-\gamma x} \frac{(\eta x)^j}{j!} e^{-\eta x} \times \frac{[\mu(t-x)]^{k-j}}{(k-j)!} e^{-\mu(t-x)} dx dA(t).$$

La matrice de transition est de la forme suivante :

$$P = \begin{pmatrix} B_{00} & A_{01} & 0 & 0 & 0 & \cdots & 0 \\ B_1 & A_1 & A_0 & 0 & 0 & \cdots & 0 \\ B_2 & A_2 & A_1 & A_0 & 0 & \cdots & 0 \\ B_3 & A_3 & A_2 & A_1 & A_0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ B_{N-1} & A_{N-1} & A_{N-2} & A_{N-3} & A_{N-4} & \cdots & A_0 \\ B_{N-1} & A_{N-1} & A_{N-2} & A_{N-3} & A_{N-4} & \cdots & A_0 \end{pmatrix},$$

avec $B_{00} = 1 - a_0 - b_0$, $A_{01} = (b_0, c_0)$.

pour tout : $0 \leq k \leq N - 1$

$$A_k = \begin{pmatrix} b_k & c_k \\ 0 & a_k \end{pmatrix},$$

et

pour tout : $1 \leq k \leq N - 1$

$$B_k = \begin{pmatrix} 1 - \sum_{i=0}^k (b_i + c_i) \\ 1 - \sum_{i=0}^k a_i \end{pmatrix}.$$

Avant de passer à la contribution majeure de e mémoire on étudie d'abord quelques performances essentielles de ce dernier.

3.2 Évaluation numérique des performances de la file GI/M/1/N avec "working vacation"

Dans cette partie, on représente la sensibilité de quelques performance, à savoir la probabilité de blocage, nombre moyenne de client dans le système et le temps moyen de séjours par rapport au capacité N et au taux moyen de service au période de "working vacation" en utilisant le logiciel Matlab mais avant on décrit les lois de ces performances. Les performance que nous allons étudier sont :

La probabilité de blocage

$$P_{loss} = \pi_{(N,0)} + \pi_{(N,1)},$$

Le nombre moyen de clients dans le système

$$L = V * \pi',$$

où $V = (0, 1, 1, 2, 2, \dots, N, N)$.

Le temps moyen de clients dans le système

$$W = \frac{L}{\lambda^*},$$

où $\lambda^* = \lambda(1 - P_{loss})$.

Notre analyse se fait pour des lois des temps inter-arrivées suivantes :

Loi Exponentielle :

$$a(t) = \lambda e^{-\lambda t};$$

Loi Hyper-Exponentielle :

$$a(t) = q\lambda_1 e^{-\lambda_1 t} + (1 - q)\lambda_2 e^{-\lambda_2 t};$$

Loi d'Erlang d'ordre 2

$$a(t) = \frac{\lambda_1 \lambda_2 (e^{-\lambda_1 t} - e^{-\lambda_2 t})}{\lambda_2 - \lambda_1};$$

Loi Déterministe de paramètre d.

3.2.1 Implémentation numérique

L'implémentation numérique sera développé sous l'environnement Matlab pour calculer les différentes performances qu'on a décrit auparavant. Ce calcul se termine par l'insertion des figures qui comparent les performances de quatre lois qu'on a vue précédemment.

Précisément, pour le calcul de performance par rapport à N , on fixe $\eta = 0.5$, $\lambda = 1$, $\gamma = 1$, $\mu = 4$. et en varie N de 3 à 10. Les résultats seront présenter dans les trois figures suivantes :

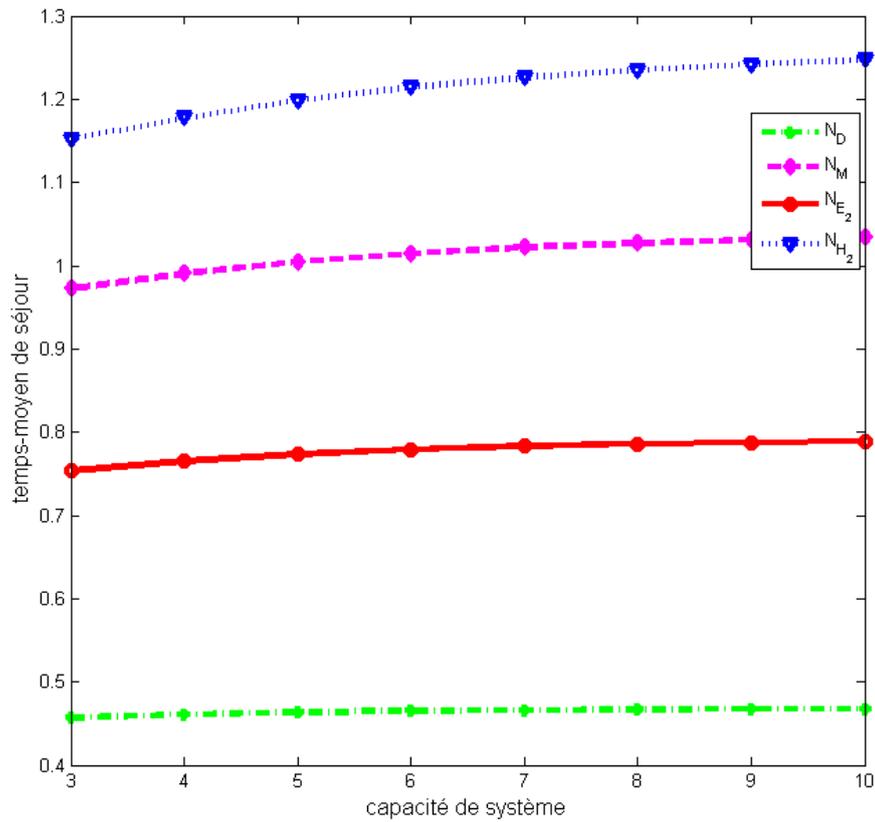


FIGURE 3.1 – Représentation graphique de temps moyen de séjour GI/M/1/N avec "working vacation".

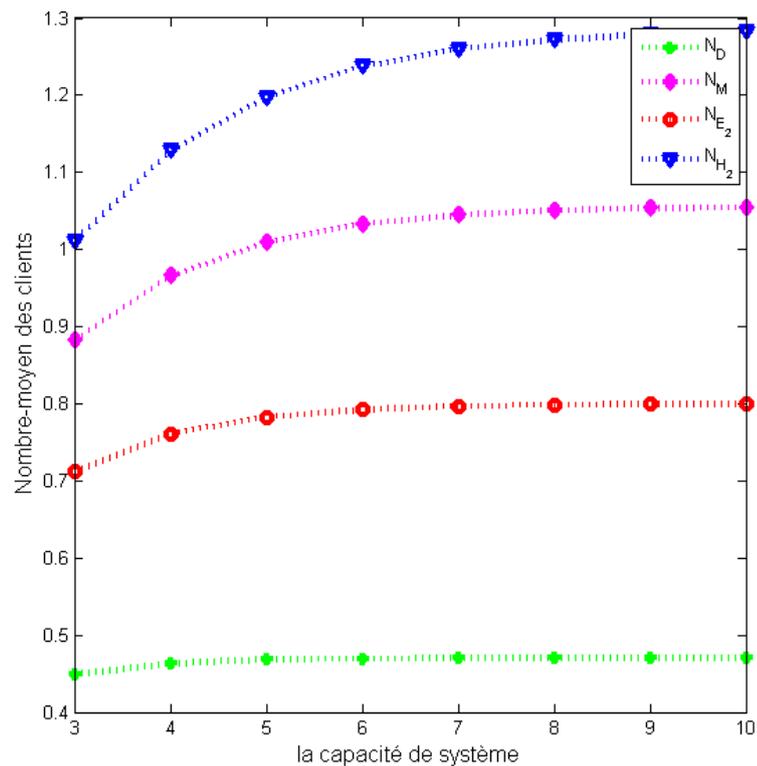


FIGURE 3.2 – Représentation graphique de nombre moyen de clients dans la file d'attente GI/M/1/N avec "working vacation".

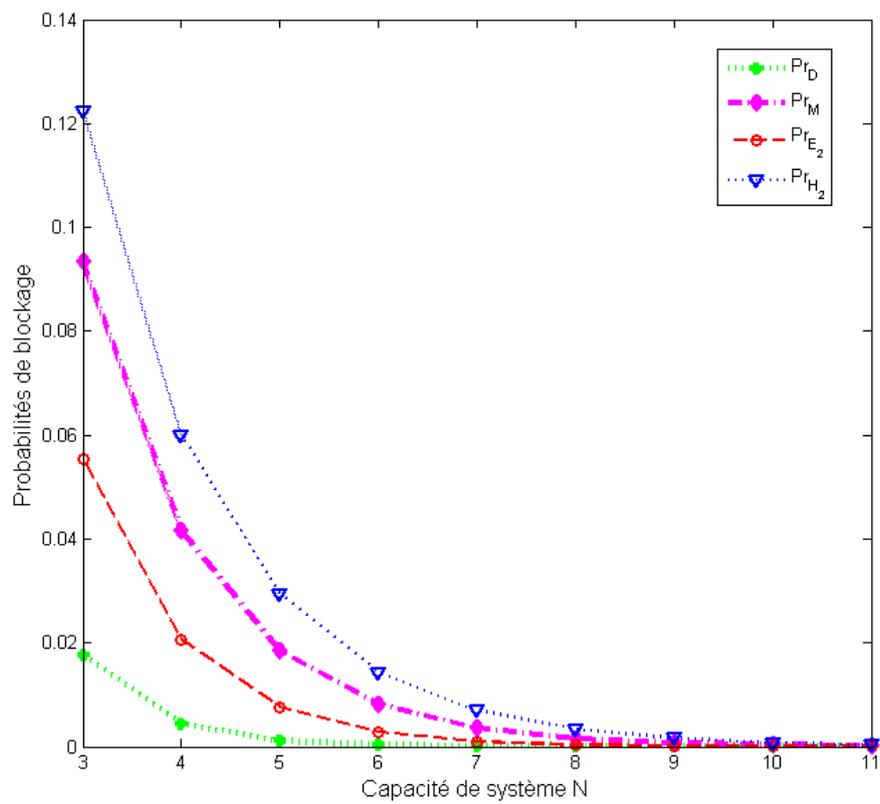


FIGURE 3.3 – Représentation graphique de probabilité de blocage de modèle GI/M/1/N avec "working vacation".

D'après les figures ci-dessus on voit que la probabilité de blocage pour les quatre lois diminuent, le temps moyen de séjour et le nombre moyen de clients dans le système augmentent par rapport à la capacité de système ce qui explique que la capacité N influence sur les performances du système. On peut déduire que l'augmentation de capacité de système est favorable pour la probabilité de blocage .

D'autre part, pour le calcul de performance par rapport à η , on fixe $N = 3$ et on varie η dans l'intervalle $[0.2, 2]$. Les résultats de ce calcul seront présentés par les trois figures suivantes :

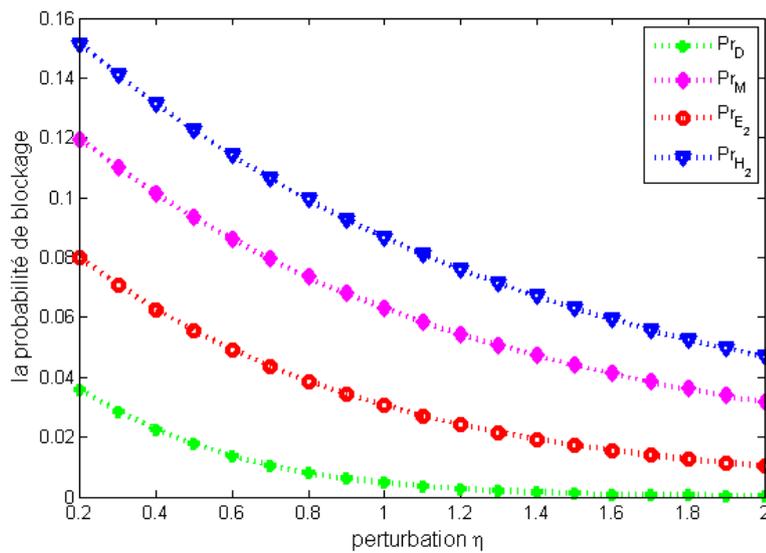


FIGURE 3.4 – Représentation graphique de probabilité de blocage dans la file d'attente GI/M/1/N avec "working vacation" en fonction de η .

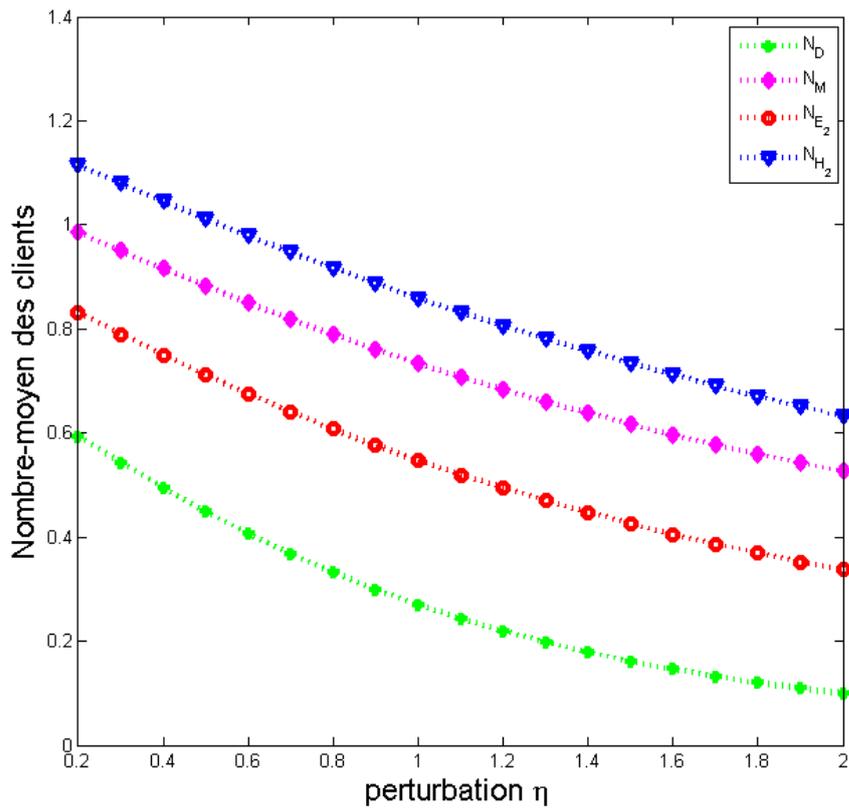


FIGURE 3.5 – Représentation graphique de nombre moyen de clients dans la file d'attente GI/M/1/N avec "working vacation" en fonction de η .

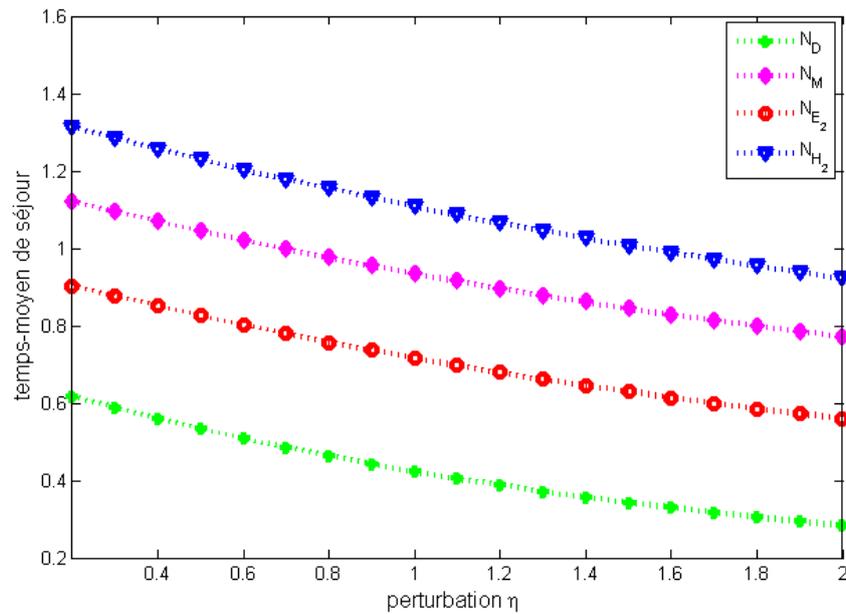


FIGURE 3.6 – Représentation graphique de temps moyen de séjour dans la file d'attente GI/M/1/N avec "working vacation" en fonction de η .

Dans ces figures on voit que les trois performances diminuent avec l'augmentation de paramètre de service η .

Dans le cas déterministe, fait un temps-moyen de séjour moins que d'autres lois ce qui implique que le serveur est plus capable de libérer le système quand les temps des inter-arrivées des clients suivent de loi déterministe, donc cela donne une probabilité de blocage moins par rapport aux autres lois et même chose pour le nombre-moyen de clients.

3.3 Description et position du problème d'optimisation

La fonction coût, que nous avons construit à minimiser par rapport au paramètre de taux de service pendant "working vacation" est de cette forme

$$\min_{\eta \in \mathbb{R}} \pi_{\eta} f \tag{3.1}$$

Ce problème d'optimisation est sans contrainte à un seul paramètre. Le but de cette partie est de chercher η^* est une solution du problème (3.1), et pour l'approximer on peut appliquer l'une des méthodes numériques d'optimisation sans contraintes qu'on a

cit  dans le deuxi me chapitre. Parmi ces m thodes on a choisi la m thode de Newton Raphson car, comme on a vu pr c demment, qu'elle converge plus rapidement que les autres m thodes, et il suffit juste de faire un bon choix pour l'approximation initiale de η pour que cette m thode soit convergente.

3.3.1 Recherche de minimum

Premi re  tape : appliquons la condition de premi re ordre

On cherche les racines de l' quation suivante :

$$(\pi_\eta f)' = 0.$$

La formule de Newton pour notre probl me est donn e par :

$$\eta_{n+1} = \eta_n - \frac{(\pi_{\eta_n} f)'}{(\pi_{\eta_n} f)''};$$

On donne   η_0 une valeur initiale puis on calcule la d riv e premi re et secondaire de la fonction objectif au point η_0 par la formule des d riv es successives de la distribution stationnaire $\pi_\eta(1.1)$ ce qui donne :

La premi re d riv e :

$$\pi_\eta^1 = \pi_\eta P' Z_\eta.$$

La deuxi me d riv e :

$$\pi_\eta^2 = \pi_\eta P_\eta^{(2)} Z_\eta + 2\pi_\eta (P^{(1)} Z_\eta)^2.$$

O  la $k^{i me}$ d riv e P est donn e par :

$$P_\eta^{(k)} = \frac{d^k}{d\eta^k} P$$

$$P_\eta^{(k)} = \begin{pmatrix} B_{00}^{(k)} & A_{01}^{(k)} & 0 & 0 & 0 & \cdots & 0 \\ B_1^{(k)} & A_1^{(k)} & A_0^{(k)} & 0 & 0 & \cdots & 0 \\ B_2^{(k)} & A_2^{(k)} & A_1^{(k)} & A_0^{(k)} & 0 & \cdots & 0 \\ B_3^{(k)} & A_3^{(k)} & A_2^{(k)} & A_1^{(k)} & A_0^{(k)} & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ B_{N-1}^{(k)} & A_{N-1}^{(k)} & A_{N-2}^{(k)} & A_{N-3}^{(k)} & A_{N-4}^{(k)} & \cdots & A_0^{(k)} \\ B_{N-1}^{(k)} & A_{N-1}^{(k)} & A_{N-2}^{(k)} & A_{N-3}^{(k)} & A_{N-4}^{(k)} & \cdots & A_0^{(k)} \end{pmatrix}$$

$$A_i^{(k)} = \frac{d^k}{d\eta^k} A_i \begin{pmatrix} \frac{d^k b_i}{d\eta^k} & \frac{d^k c_i}{d\eta^k} \\ 0 & \frac{d^k a_i}{d\eta^k} \end{pmatrix}$$

pour $i \geq 0$:

$$B_i^{(k)} = \begin{pmatrix} -\sum_{i=0}^k \left(\frac{d^k}{d\eta^k} b_i + \frac{d^k}{d\eta^k} c_i \right) \\ -\sum_{i=0}^k \frac{d^k}{d\eta^k} a_i \end{pmatrix}$$

La fonction coût choisie dans ce travail est la suivante :

$$\pi_\eta f = \pi_\eta (c_1 L + c_2 P_{bs}^1 + c_3 P_{wv}^0 + c_4 P_{loss}),$$

où

c_1 : le coût par unité de temps pour chaque client présent dans le système.

c_2 : le coût par unité de temps pour le service pendant la période de service normale.

c_3 : le coût par unité de temps pour le service pendant la période de working-vacation.

c_4 : le coût fixe pour chaque client perdu lorsque le système est bloqué.

On peut écrire f de cette manière

$$\pi_\eta f = \pi_\eta (c_1 V_L + c_2 V_{P_{bs}^1} + c_3 V_{P_{wv}^0} + c_4 V_{P_{loss}}),$$

Où

$$V_L = (0, 1, 1, 2, 2, \dots, N, N),$$

$$V_{P_{bs}^1} = (0, 0, 1, 0, 1, \dots, 0, 1),$$

$$V_{P_{wv}^0} = (1, 1, 0, 1, 0, \dots, 1, 0),$$

$$V_{\text{loss}} = (0, 0, 0, 0, 0, \dots, 1, 1).$$

On note

$$V_c = c_1 V_L + c_2 V_{P_{bs}^1} + c_3 V_{P_{wv}^0} + c_4 V_{\text{loss}}.$$

En effet la fonction objectif est donnée par :

$$\pi_\eta f = V_c \pi_\eta.$$

Dans notre étude on a réussi à trouver un minimum pour le modèle décrit par une loi des temps des inter-arrivées qui suivent la loi exponentielle. Le minimum est trouver pour les valeurs de modèle suivante : ($N = 3, \mu = 4, \gamma = 1, \lambda = 1, \eta_0 = 0.01, c_1 = 1, c_2 = 2, c_3 = 8, 5, c_4 = 1$) et on a pris $\epsilon = 10^{-14}$. Les résultats sont présentés dans le tableau suivant :

n	η
0	0.0100000000000000
1	0.434378621171999
2	0.560739974128732
3	0.573844900245524
4	0.573979709675120
5	0.573979723839688
6	0.573979723839676
7	0.573979723839685

La solution minimal se trouve au point $\eta^* = 0.573844900245524$ avec une valeur minimale 8.292485188895277 pour la fonction coût.

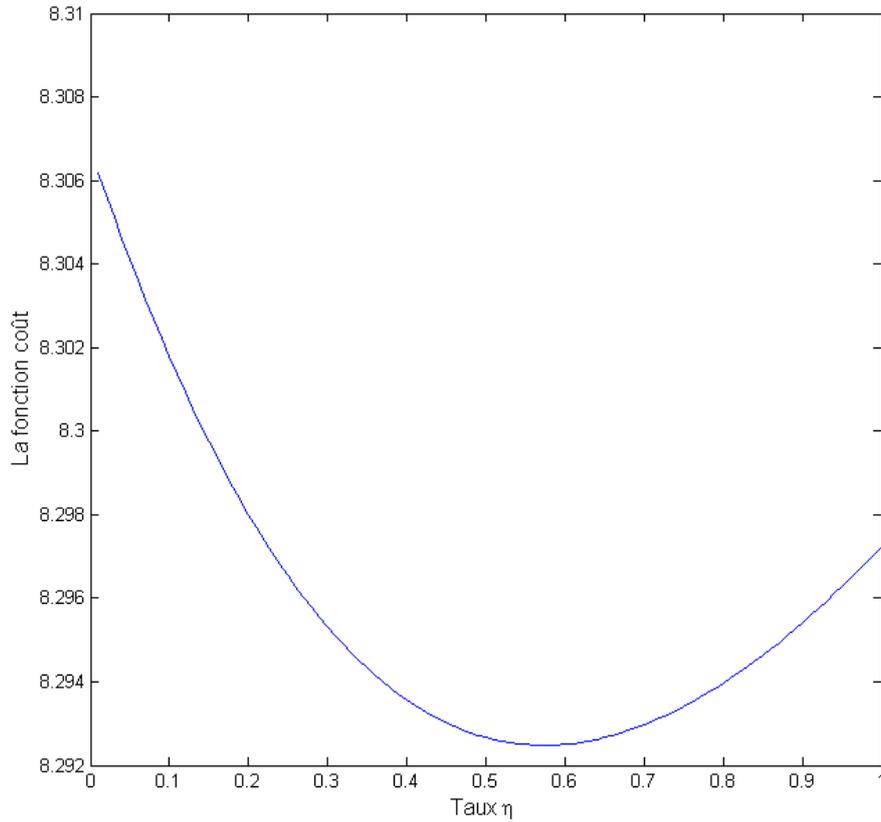


FIGURE 3.7 – Représentation graphique de la fonction du coût.

Conclusion

La convergence de la méthode de Newton vers le minimum après la quatrième itération implique la rapidité de cet algorithme pour trouver un minimum d'une fonction ; en plus d'après le résultat obtenu de minimum on peut conclure que cet algorithme peut calculer efficacement la valeur presque optimale de la fonction de coût total.

Conclusion générale

Dans ce mémoire, nous nous sommes intéressés à un système de files d'attente avec la politique de multiple "working vacation" car c'est une politique très importante, elle est utilisée par plusieurs systèmes, en plus elle peut satisfaire les clients par rapport au autres politiques comme la politique de vacance. Le modèle qu'on a traité avec cette politique est le modèle de file d'attente GI/M/1/N à une capacité finie et notre étude est comme suite :

Dans le premier chapitre on a fait un rappel sur la description mathématique et les mesures de performances d'une file d'attente qu'on a besoin pour l'utiliser dans l'analyse de notre modèle.

Dans le deuxième chapitre on a fait un rappel sur l'optimisation sans contrainte et on a cité quelques méthodes d'optimisation .

Dans le troisième chapitre on a calculé quelques mesures de performances pour le modèle GI/M/1/N avec "working-vacation" à savoir la probabilité de blocage, le temps moyen de séjour et le nombre moyen de clients dans le système par rapport au capacité de système et au taux moyen de service pendant la période de "working vacation". Dans ce travail on est arrivé à faire une optimisation pour ce modèle on a trouvé qu'on peut minimiser les coûts de système GI/M/1/N par rapport au taux moyen de service pendant la période de "working vacation".

En terme de continuité de ce travail, plusieurs perspectives de recherche peuvent-être envisagées, on peut citer :

- Appliquer le même procédé pour d'autre système de file d'attente.
- Elargir l'applicabilité de la même approche à plusieurs variables.
- Considerer la même problématique dans le cas aléatoire (optimisation robuste).

Bibliographie

- [1] A. Aissani, S. Taleb, T. Kernane, G. Saidi and D. Hamadouche, An M/G/1 retrial queue with working vacation, *Advances in Intelligent Systems and Computing*, 240 (2014),443-452.
- [2] D. Arivudainambi, P. Godhandaraman and P. Rajadurai, Performance analysis of a single server retrial queue with working vacation, *OPSEARCH*, 51 (2014), 434-462.
- [3] Y. Baba, Analysis of a GI/M/1 queue with multiple working vacations, *Operations. Research. Letters*, 33 (2005), 201-209.
- [4] Y. Baba, The MX/M/1 queue with multiple working vacations, *Am. J. Operation. Research*, 2 (2012), 217-224.
- [5] A.D. Banik, U.C. Gupta and S.S. Pathak, On the GI/M/1/N queue with multiple working vacations analytic analysis and computation, *Applied. Mathematical. Modelling.*, 31(2007), 1701-1710.
- [6] H.Y. Chen, J.H. Li. and N.S. Tian, The GI/M/1 queue with phase-type working vacations and vacation interruption, *J Applied. Mathematical. Computer*, 30 (2009), 121-141.
- [7] T.V. Do, M/M/1 retrial queue with working vacations, *Acta Informatica*, 47 (2010), 67-75.
- [8] B.T. Doshi, Queueing systems with vacations : a survey, *Queueing System*, 1 (1986), 29-66.
- [9] S. Gao and Z. Liu, An M/G/1 queue with single vacation and vacation interruption under Bernoulli schedule, *Applied Mathematical Modelling*, 37 (2013), 1564-1579.
- [10] S. Gao, J. Wang and W. Li, An M/G/1 retrial queue with general retrial times, working vacations and vacation interruption, *Asia-Pacific Journal of Operational Research*, 31 (2014), 6-31.

-
- [11] S. Gao and Y. Yao, An MX/G/1 queue with randomized working vacations and at most J vacations, *International Journal of Computer Mathematics* 91 (2014), 368-383.
- [12] V. Goswami, P. Vijayalaxmi and K. Jyothsna, Analysis of GI/M(n)/1/N queue with state-dependent multiple working, *OPSEARCH*, 50 (2013), 106-124.
- [13] G. laurent, optimisation sans contrainte de fonctions continues non linéaire ;2012.21,22
- [14] M. Jain and A. Jain, Working vacations queueing model with multiple types of server breakdowns, *Applied Mathematical Modelling*, 34 (2010), 1-13.
- [15] J.C. Ke, C.H. Wu and Z.G. Zhang, Recent developments in vacation queueing models : A short survey, *International Journal of Operations Research*, 7 (2010), 3-8.
- [16] J. Kim, B.D. Choi, and K. Chae, Analysis of queue-length distribution of the M/G/1 queue with working vacation. *International. Conference. on Statistics and Related fields*, Hawaii (2003).
- [17] D.H. Lee and B.K. Kim, A note on the sojourn time distribution of an M/G/1 queue with a single working vacation and vacation interruption, *Operations Research Perspectives*, 2 (2015), 57-61.
- [18] Y. Levy and U. Yechiali, Utilization of idle time in an M/G/1 queueing system, *Management Science*, 22 (1975), 202-211.
- [19] J. Li, and N. Tian, The discrete-time GI/Geo/1 queue with working vacations and vacation interruption, *Applied. Mathematical. Computer*, 185 (2007), 1-10.
- [20] J. Li, and N. Tian, The M/M/1 queue with working vacations and vacation interruptions, *J Syst Sci Syst Eng*, 16 (2007), 121-127.
- [21] J. Li, N. Tian and Z. Ma, Performance analysis of GI/M/1 queue with working vacations and vacation interruption, *Applied. Mathematical. Modelling*, 32 (2008) 2715-2730.
- [22] J. Li, N. Tian, Z.G. Zhang, and H.P. Luh, Analysis of the M/G/1 queue with exponentially working vacations a matrix analytic approach, *Queueing Syst.*, 61 (2009), 139-166.
- [23] J. Li, and N. Tian, Performance analysis of a GI/M/1 queue with single working vacation, *Applied. Mathematical. Computer*, 217 (2011) 4960-4971.
- [24] P. Li, X. Xu, and X. Cao, Analysis for the M/G/1 working vacation queue with set-up period, *Journal of Information and Computational Science*, 9 (2012), 801-812.
-

- [25] T. Li, Z. Wang and Z Liu, Geo/Geo/1 retrial queue with working vacations and vacation interruption, *J. Applied. Mathematical. Computer*, 39 (2012), 131-143.
- [26] C. Lin and J.C. Ke, Multi-server system with single working vacation, *Applied. Mathematical Modelling*, 33 (2009), 2967-2977.
- [27] B. Liu, L. Cui, Y. Wen, and J. Shen, A cold standby repairable system with working vacations and vacation interruption following Markovian arrival process, *Reliability Engineering. System. Safety*, 142 (2015), 1-8.
- [28] W. Liu, X. Xu, and N. Tian, Stochastic decompositions in the M/M/1 queue with working vacations, *Operational. Research. Letters.*, 35 (2007), 595-600.
- [29] Z. Liu and Y. Song, Geo/Geo/1 retrial queue with non-persistent customers and working vacations. *Journal. Applied. Mathematical. Computer*, 42 (2013), 103-115.
- [30] S.Ouazine and K.Abbas. *Applied Mathematical Modelling*, 2016.
- [31] P. Rajadurai, M.C. Saravananarajan, and V.M. Chandrasekaran, Analysis of an M/G/1 retrial queue with balking, negative customers, working vacations and server breakdown, *International. Journal. Applied. Engineering. Research*, 10 (2015), 4130-4135.
- [32] P. Rajadurai, S. Yuvarani and M.C. Saravananarajan, Performance analysis of preemptive priority retrial queue with immediate Bernoulli feedback under working vacations and vacation interruption, *Songklanakarin Journal. Sci. Technol*, (2016)-Accepted.
- [33] N. Selvaraju and C. Goswami, Impatient customers in an M/M/1 queue with single and multiple working vacations, *Computers and Industrial Engineering*, 65 (2013), 207-215.
- [34] L.D. Servi, and S.G. Finn, M/M/1 queues with working vacations, *Performance. Evaluation*, 50 (2002), 41-52.
- [35] C. Sreenivasan, R. Srinivas, A. Chakravarthy and A. Krishnamoorthy, MAP/PH/1 queue with working vacations, vacation interruptions and N policy, *Applied Mathematical Modelling*, 37 (2013) 3879-3893.
- [36] S. Upadhyaya, Working vacation policy for a discrete-time GeoX/Geo/1 retrial queue, *OPSEARCH*, 52 (2015), 650-669
- [37] H. Takagi, *Queueing analysis : A foundation of performance evaluation (Vol. 1)*. North- Holland, (1991).

-
- [38] L. Tao, Z. Liu, and Z. Wang, The GI/M/1 queue with start-up period and single working vacation and Bernoulli vacation interruption, *Applied. Mathematical. Computer*, 218 (2011), 4401- 4413.
- [39] L. Tao, Z. Liu, and Z. Wang. M/M/1 retrial queue with collisions and working vacation interruption under n-policy. *RAIRO - Operations Research*, 46 (2012), 355-371,
- [40] N. Tian, and Z.G. Zhang. *Vacation queueing models : Theory and applications*. New York : Springer-Verlag (2006).
- [41] N. Tian, X. Zhao and K. Wang. The M/M/1 queue with single working vacation, *International Journal of Information Management Sciences*, 19 (2008), 621-634.
- [42] N. Tian, J. Li, and Z.G. Zhang, Matrix analytic method and working vacation queues A survey, *Int. Journal of Information and Management Sciences*, 20 (2009), 603-633.
- [43] P. Vijayalaxmi, and P. Rajesh, Analysis of variant working vacations on batch arrival queues, *OPSEARCH*, (2015). DOI 10.1007/s12597-015-0236-3.
- [44] P. Vijaya Laxmi and K. Jyothsna, Performance analysis of variant working vacation queue with balking and renegeing, *International. Journal. of Mathematical. Operational. Research*, 6 (2014).
- [45] P. Vijaya Laxmi, V. Goswami and V. Suchitra, Analysis of GI/M(n)/1/N queue with single working vacation and vacation interruption, *International. Journal. Mathematical, Computer, Physical., Electrical, Computer. Engineering.*, 7 (2013), 740-746.
- [46] P. Vijayalaxmi and V. Suchitra, Analysis of General Input State Dependent Working Vacation Queue with Changeover Time, *International Scholaly Reaserch Notices : Computational Mathematics*, 2014 (2014), 1-8
- [47] D. Wu, and H. Takagi, M/G/1 queue with multiple working vacations, *Performance. Evaluation.*, 63 (2006), 654681.
- [48] X. Xu, Z. Zhang and N. Tian, Analysis for the Mx/M/1 working vacation queue, *Information. Management. Sciences.*, 20 (2009), 379-394.
- [49] D.H. Yang, and D. Wu, Cost-minimization analysis of a working vacation queue with N- policy and server breakdowns, *Computers and Industrial Engineering*, 82 (2015), 151-158.

- [50] Z. Zhang, and X. Xu, Analysis for the M/M/1 queue with multiple working vacations and N-policy, *International Journal of Information and Management Sciences*, 19 (2008), 495-506.
- [51] M. Zhang, and Z. Hou, Performance analysis of M/G/1 queue with working vacations and vacation interruption, *J Computer Applied Mathematical*, 234 (2010), 2977-85.
- [52] M. Zhang, and Z. Hou, Steady state analysis of GI/M/1/N queue with a variant of multiple working vacations, *Computers. Industrial. Engineering*. 61 (2011), 1296-1301.
- [53] M. Zhang, and Z. Hou, Performance analysis of MAP/G/1 queue with working vacations and vacation interruption, *Applied Mathematical Modelling*, 35 (2011) 1551-1560.
- [54] M. Zhang, and Z. Hou, M/G/1 queue with single working vacation, *Journal of Applied. Mathematical. Computers.*, 39 (2012), 221-234
- [55] M. Zhang and Q. Liu, An M/G/1 G-queue with server breakdown, working vacations and vacation interruption. *OPSEARCH*,52 (2015), 256-270
- [56] G. Zhao, X. Du and N. Tian, GI/M/1 Queue with Setup Period and Working Vacation and Vacation Interruption, *International. Journal. Information. Management. Sciences.*, 20 (2009), 351-363.
- [57] G. Zhao, and G. Cui, Approximation on MAP/G/1 Queue with N-policy Working Vacation, *Journal. Conversion. Information. Technical*, 6 (2011), 110-118.
- [58] Kemeny, J.G. and Snell, J.L. *Finite Markov Chains*, Van Nostrand, New York, 1960.
- [59] Keilson, J. *Markov Chain Models-Rarity and Exponentiality*. Springer-Verlag, New York, 1979.

Résumé :

Notre travail est basé sur l'étude de l'optimisation d'une fonction coût d'un modèle de file d'attente GI/M/1/N avec "working vacation"; on a cherché à optimiser cette fonction par rapport au taux moyen de service pendant "working vacation". Dans cette étude, on a utilisé l'algorithme de Newton Raphson. En outre, nous avons effectué l'analyse de sensibilité des performances du modèle étudié par rapport à la variation de capacité de système et le taux moyen de service pendant la période de "working vacation". Plusieurs exemples numériques ont été présentés.

Mots clés: File d'attente avec "working vacation", chaîne de Markov, Optimisation, Méthode de Newton Raphson.

Abstract:

Our work is based on the study of the optimization of a function of the cost of the GI/M/1/N queue with "working vacation"; we search to optimize this function with respect to of service rate during "working vacation". In this study we used the algorithm of Newton Raphson. Moreover, we analyze the sensitivity analysis of the model performances studied against the system capacity variation and the average service rate during the working vacation period. Several numerical examples are presented.