



République Algérienne Démocratique et Populaire Ministère de
l'enseignement supérieur et de la recherche scientifique

Université Larbi Tébessi - Tébessa

Faculté des Sciences Exactes et des Sciences de la Nature et de la Vie

Département : Mathématiques et Informatique



Mémoire de fin d'études

Pour l'obtention du diplôme de MASTER

Domaine : Mathématiques et Informatique

Filière : Informatique

Option :

Thème

**Nouvel algorithme hybride d'ordonnancement des tâches
avec logique flou dans le CLOUD COMPUTING.**

Présenté Par : SELIM Nour Djihen

Devant le jury :

Pr. LAOUER Mohamed Redha	Pr	Université Larbi Tébessi	Président
Dr. MERZOUG Soltan	MCB	Université Larbi Tébessi	Examineur
Dr ALI Widad	MCB	Université Larbi Tébessi	Encadreur
Dr. BOUAKKAZ Fatima	MAA	Université Larbi Tébessi	Co- Encadreur

Date de soutenance : Juin 2022

Résumé

En quelques années, le Cloud Computing est devenu le modèle commercial informatique le plus prometteur. Grâce à ses différents atouts techniques et financiers, le Cloud computing continue de convaincre chaque jour de nouveaux utilisateurs issus des secteurs scientifiques et industriels. Pour satisfaire les différents besoins des utilisateurs, les fournisseurs de Cloud doivent maximiser la performance de leurs ressources pour assurer le meilleur service au moindre coût. Les efforts d'optimisation des performances dans le Cloud peuvent être réalisés à différents niveaux et aspects. Dans le présent travail, nous proposons d'introduire un processus de logique floue dans la stratégie d'ordonnancement du Cloud afin d'améliorer le temps de réponse, le temps de traitement et le coût total. En fait, la logique floue a prouvé sa capacité à résoudre le problème de l'optimisation dans plusieurs domaines tels que l'exploration de données, le traitement d'images, la mise en réseau et bien plus encore.

ملخص

مع مرور الوقت، أصبحت الحوسبة السحابية نموذج الأعمال الواعد لتكنولوجيا المعلومات. بفضل مزاياها الفنية والمالية النافعة المتنوعة، حيث تواصل الحوسبة السحابية إثباتها للمستخدمين الجدد كل يوم من القطاعات العلمية والاقتصادية. أنها تلي جميع الاحتياجات المختلفة لهم، لذا يجب على موفري السحابة الإلكترونية تحسين أداء مواردهم لضمان أفضل خدمة بأقل تكلفة. ويمكن ذلك ببذل جهود تحسين أداء السحابة على مستويات وجوانب مختلفة. في هذه المذكرة، نقترح إدخال عملية المنطق الترجيحي على لوغاريتمية تخدم إستراتيجية الجدولة السحابية لتحسين وقت الاستجابة ووقت المعالجة والتكلفة الإجمالية. وأثبتنا أنّ المنطق الترجيحي أثبت قدرته على حل مشكلة التحسين في العديد من المجالات مثل تنقيب البيانات ومعالجة الصور والشبكات وغير ذلك.

Abstract

Within a few years, Cloud Computing has become the most promising IT business model. Thanks to its various technical and financial advantages, cloud computing continues to convince new users every day from the scientific and industrial sectors. To meet the different needs of users, cloud providers must maximize the performance of their resources to ensure the best service at the lowest cost. Cloud performance optimization efforts can be made at different levels and aspects. In the present work, we propose to introduce fuzzy logic process in cloud scheduling strategy to improve response time, processing time and total cost. In fact, fuzzy logic has proven its ability to solve the problem of optimization in several fields such as data mining, image processing, networking and much more.

Remerciement

Nous remercions ALLAH Tout Puissant, d'abord et avant tout, de nous avoir donné la force et patience et surmonter les difficultés. pour terminer ce que nous avons commencé dans le voyage d'étude . Nous demandons à ALLAH de nous aider dans ce qui reste de cette vie Je tiens à exprimer mes profonds

Remerciements à mon encadreuse ALLI Wided , pour les encadrements, la confiance et pour le soutien qu'elles m'ont apportée durant tout ce travail. Leur compétence scientifique a toujours été source d'enrichissement qui me permettant de faire à bien ce travail. Merci pour tous vos conseils et votre patience.

Un grand merci à tous les membres du Jury qui me font l'honneur d'accepter de juger mon travail. Je remercie tous les enseignants et les étudiants du département d'informatique de l'Université de Tébessa.

Je voudrais aussi présenter mes remerciements à mon marie et ma mère et mes frères pour tous leurs sacrifices en faveur de nos éducations, mes amis pour leur support qui j'ai été autorisé à de travailler dans les meilleures conditions

DEDICACE

À l'homme , qui me conseille , m'encourage et me soutenir tout au long de mes études.

Mon Chère Marie KLAË Karim.

À l'homme , mon précieux offre du dieu , qui doit ma vie , ma réussite et tout mon respect.

Mon Père SELIM Ramdane .

À la femme qui a souffert sans me laisser souffrir, qui n'a jamais dit non à mes exigences et qui n'a épargné aucun effort pour me rendre heureuse.

Mon adorable mère GAHFEZ Laïla.

À la fille qui sait toujours comment procurer la joie et le bonheur pour toute la famille.

Madorable petite sœur SLIM Anfel.

À mes frères qui aidés dans les moments difficiles.

À tous les amis que j'ai connus à maintenant . Merci pour leurs amours et leur encouragement.

À vous tous

Je dédie ce travail.

Table des matières

Remerciement	III
<i>DEDICACE</i>	IV
Table des matières	V
Liste des figures.....	VI
Liste des Tableaux	VII
Liste des abbreviations	VIII
Liste des Algorithmes.....	IX
Introduction Générale	1
Chapitre I : Généralité sur les CLOUD Computing	
I.1. Introduction	4
I.2. Définitions	5
I.2.1. Terminologie « Cloud »	6
I.3. Architecture d'un système cloud.....	6
I.4. Caractéristiques essentielles.....	7
I.5. Les Services de Cloud Computing.....	8
I.6. Cloud technologies et infrastructure.....	11
I.6.1. Périmètre de réseau logique (Logical Network Perimeter)	11
I.6.2. Serveur virtuel	11
I.6.3. Périphérique de stockage en nuage.....	12
I.6.4. Moniteur d'utilisation du cloud.....	12
I.6.5. Réplication des ressources.....	13
I.6.6. Environnement prêt à l'emploi (Ready-Made Environment)	13
I.7. Modèles de déploiement.....	13
I.8. Challenges de recherche en environnement de cloud computing	14
II.8.1. Sécurité, confidentialité et confiance	14
II.8.2 Verrouillage des données et standardisation.....	14

II.8.3. Disponibilité, tolérance aux pannes et reprise après des échecs	14
II.8.4. Gestion des ressources et efficacité énergétique.....	15
I.9. Avantages et désavantage de Cloud computing	15
I.9.1. Avantages du Cloud computing	15
A. Avantages au niveau de la stratégie	15
B. Avantages au niveau des fonctions et des processus métier	15
C. Avantages opérationnels	16
I.9.2. Inconvénients du Cloud computing.....	16
I.10. Clouds Exemples	16
I.10.1. Google (Google AppEngin)	16
I.10.2. EMC	17
I.10.3. Microsoft	18
I.10. Conclusion.....	20

Chapitre II : l'ordonnement dans les Clouds Computing

II.1. Introduction.....	22
II.2. Présentation des algorithmes d'ordonnement des tâches	23
II.2.1. Définitions.....	23
II.2.2. Notions lies à l'ordonnement.....	23
II.3. Système de planification de tâches dans le cloud computing	24
II.3.1. Niveaux d'ordonnement	25
II.3.2. Avantages des algorithmes de planification de tâches.....	25
II.4. Classifications des algorithmes d'ordonnement des tâches	25
II.5. Combiner les stratégies d'ordonnement (Hybridation des algorithmes).....	29
II.6. Problèmes, algorithmes et complexité de l'ordonnement	31
II.7. Contraintes des stratégies d'ordonnement [20].....	32
II.8. Ordonnement de temps d'exécution et les conditions du déclenchement des tâche.....	32
II.9. Exemple d'une stratégie d'ordonnement.....	33

II.9.1. Amazone	33
II.10. Problèmes de l’ordonnancement dans le cloud.....	33
II.11. Optimisation et résoudre les problèmes de l’ordonnancement dans les clouds avec logique flou	34
II.11.1. Etat de l’art et travaux connexes	34
II.11.2. Notions sur la logiques floue	36
A) Logique floue.....	36
II.11. Application de logique floue dans les problèmes de l’ordonnancement	36
II.12. Conclusion	39
Chapitre III: Conception d'un ordonnanceur basé sur la logique floue dans le Cloud	
III.1. Introduction.....	41
III.2. Notre contribution.....	41
III.2.1. Problématique	41
III.2.2. Objectifs.....	41
III.2.3. Modèle proposé	42
III.2.4. Caractéristiques de modèle proposé	44
III.3. La méthode d'ordonnancement proposée avec la logique floue	44
III.4. Conclusion	48
Chapitre IV: Implémentation d’un algorithme d’ordonnancement hybridée avec logique floue	
IV.1. Introduction	50
IV.2. Outils de développement	51
IV.2.1 Environment machine.....	51
IV.2.2. Environnement logiciel	51
IV.3. Architecture générale.....	51
IV.4. Métriques de performances	53
IV.5. Résultats de simulation.....	53
IV.6. Conclusion.....	57

Conclusion Générale.....	59
Bibliographie.....	61

Liste des figures

Chapitre I

Fig I. 1: exemple d'architecture générale d'un cloud	7
Fig I. 2: Cloud Framework (services) selon les niveaux d'abstract.....	9

Chapitre II

Fig II. 1: Schéma d'ordonnancement simple	24
Fig II. 2: Vu schématique de problème d'ordonnancement	31
Fig II. 3: Structure du contrôleur de logique floue	38

Chapitre III

Fig III. 1: Schéma générale de système proposé	43
Fig III. 2: les entrés et les sorties de système de logique floue	45
Fig III. 3: Diagramme d'activité de système de logique floue proposé.....	47

Chapitre IV

Fig. IV 1: Architecture CloudSim pour l'ordonnancement.....	52
Fig. IV 2: Exemple d'exécution de simulation 1	53
Fig. IV 3: Simulation Cloudsim Pour l'ordonnancement	56

Liste des Tableaux

Chapitre II

Tableau II. 1 : Analyse comparative des principales stratégies de planification.....	27
Tableau II. 2 : Compatibilité (1) ou incompatibilité (0) des relations de la stratégie de planification intégrée de base	30
Tableau II. 3 : :Les études académiques connexes.....	35

Chapitre III

Tableau III. 1: les règles utilisées dans le système de logique floue.....	46
--	----

Chapitre IV

Tableau IV 1: le Makspan (en nanosecondes) avec différents nombres of cloudets et VM en utilisant FCFS.....	54
Tableau IV 2: le Makspan (en nanosecondes) avec différents nombres of cloudets et VM en utilisant l'algorithme proposé	55

Liste des abbreviations

APaaS	Application Platform-as-a-Service
API	Application Programming Interface
DaaS	Desktop as a Service
FaaS	Framework-as-a-Service
FIS	Fuzzy interference system
IaaS	Infrastructure-as-a-Service
IDE	Integrated development environment
LAN	Local Area Network
NIST	National Institute of standards and technologies
PaaS	Platform-as-a-Service
PnP	plug-and-play
SaaS	Software-as-a-Service
SLA	Service Layer Accord
VLAN	Virtual Local Area Network
VM	Virtual Machine

Liste des Algorithmes

FCFS	First Come First Serve
LB	Load Balance
PSO	Particle Swarm Optimization
RF	Random Forest
RR	Round Robin

Introduction générale

Introduction Générale

Contexte

Le problème d'ordonnancement se pose dans d'innombrables domaines, et il évolue au fil du temps avec l'industrie et la technologie. [1] Avec le développement des ordinateurs, l'ordonnancement dans les processeurs informatiques a fait l'objet d'une grande attention, l'objectif le plus courant étant la minimisation des temps d'exécution des tâches, également connu sous le nom de Makespan. Outre quelques particularités, les principes de base sont restés les mêmes que dans l'ordonnancement des activités entre les machines en production. Dans les supercalculateurs, l'ordonnancement multiprocesseur considère plusieurs processeurs parallèles de même capacité. De plus, la source de données est considérée comme centralisée et connectée par un canal à haut débit entre les processeurs, de manière à ce que les activités (ou les emplois) puissent échanger des messages rapidement. [2]

Plus récemment, le Cloud computing propose des ressources informatiques, souvent virtualisées, comme des services aux utilisateurs, cachant les aspects techniques de gestion des ressources [5]. Par conséquent, les clusters et les grilles peuvent faire partie des centres de données dans l'infrastructure de cloud computing, exigeant de nouveaux objectifs d'optimisation et des variables communes à l'informatique verte [6] et à l'informatique utilitaire [7]. Kwok et Ahmad ont déclaré en 1999 [3] que la prise en compte de plates-formes hétérogènes était une direction difficile pour étendre les algorithmes d'ordonnancement. Suite à la vulgarisation de ces plates-formes dans les grilles et le cloud computing, de nouveaux concepts d'ordonnancement sont apparus dans la littérature [8]. Alors que d'une part les aspects fondamentaux de l'ordonnancement restent inchangés, d'autre part, différents objectifs d'optimisation ont gonflé la littérature sur l'ordonnancement au cours de la dernière décennie. Une telle houle dans le domaine a apporté tellement d'informations qu'il est devenu difficile de reconnaître la contribution exacte des nouveaux résultats.

Étant donné que les défis de l'ordonnancement existent toujours, Smith soutient dans [13] que l'ordonnancement n'est pas un problème entièrement résolu, il a déclaré que " des techniques d'ordonnancement qui tiennent correctement compte de l'incertitude, permettent un changement de solution contrôlé et soutiennent une négociation et un raffinement efficaces des contraintes sont cruciaux conditions préalables, et la nécessité d'opérer dans le contexte de multiples agents intéressés est une donnée."

Le contrôleur flou proposé utilise la logique floue, introduite par Zadeh en 1965 [10]. La logique floue n'a pas d'affectation stricte d'éléments à des ensembles comme la logique

binaire. Au lieu de cela, chaque élément a un degré d'appartenance à un ensemble. Ce degré est représenté par une valeur comprise entre 0 et 1. Pour pouvoir appliquer la logique floue à un problème spécifique tel que l'ordonnancement entre cloudlets ou machines virtuelles, un système flou doit être construit. La construction se déroule en trois étapes : (Fuzzyfication, Interference engine, De-Fuzzyfication). Le processus de conception d'un système de contrôle flou consiste en une série d'étapes. La première étape du contrôle flou consiste à définir les variables d'entrée et les variables de contrôle. Chaque variable doit être quantifiée. Ensuite, chaque quantification de la variable se voit attribuer une fonction d'appartenance. Ensuite, une base de règles floues doit être conçue, cette base de règles détermine quelle action de contrôle a lieu dans quelles conditions d'entrée. Les règles sont écrites dans un format si-alors. Une formule d'implication est utilisée pour évaluer les règles individuelles si-alors dans la base de règles [11].

Contribution

Notre contribution a repris dans l'adaptation de la théorie floue comme solution pour améliorer les services cloud à travers les tâches d'ordonnancement, en implémentant une logique d'interférence floue dans un algorithme FCFS classique et en comparant les résultats avec un FCFS purement fait.

Notre algorithme optimise le Makespan, c'est-à-dire le temps d'exécution du workflow qui est la métrique la plus importante lorsqu'on s'intéresse à l'ordonnancement de tâches.

Organisation de la mémoire

Dans le présent travail nous proposons un modèle d'optimisation basé sur la logique floue adressé au niveau d'ordonnancement de tâche dans le cloud computing. Ce travail s'organise en 04 chapitres. Le premier chapitre dresse une généralité sur les Clouds et Grid Computing, L'architecture et Conception fondamentaux du fonctionnement et aussi présente la littérature qui couvre les définitions de base concernant. et en terminant le chapitre par un exemple fameux d'un Cloud.

Le chapitre 2 présente les définitions nécessaires pour l'ordonnancement des tâches dans les clouds, les différents types d'ordonnancement et les Challenges nuisant à leurs performances. Chapitre 03 est dédié à la conception de notre approche proposée. Le chapitre 4 attribué à l'implémentation et l'environnement de développement adopté. Dans un premier temps, nous introduisons le simulateur utilisé CloudSim. En deuxième lieu nous intégrons notre algorithme, et on a terminé par réalisation, exécution et analyse des résultats.

Finalement, nous clôturons ce mémoire par une conclusion générale.

Chapitre I :
Généralité sur les CLOUD Computing

I.1. Introduction

Le Cloud Computing s'impose très rapidement comme étant un standard pour l'hébergement des applications et les services logiciels. La plupart des entreprises, individus et même des corps gouvernementaux se réfugient vers le Cloud en raison de plusieurs avantages que ce nouveau paradigme offre, y compris la réduction des coûts, l'évolutivité rapide, la facilité de développement, le stockage illimité, et l'accessibilité omniprésente. Plusieurs applications dans de nombreux domaines contiennent généralement de nombreuses tâches. Ces tâches requièrent pour leurs exécutions sur le Cloud un ordonnancement, c'est l'affectation des tâches aux ressources disponibles sur la base des exigences et les caractéristiques des tâches. L'ordonnancement est un processus très important pour un fonctionnement efficace du Cloud.

Ce chapitre présente un état de l'art sur la technologie du Cloud Computing, Tout d'abord, nous donnons la définition du cloud et à quoi ça sert le cloud. Ensuite, il présente les caractéristiques essentielles du Cloud computing , aussi ses technologies , ses services et ses challenges de recherche. Le chapitre se termine par introduire les avantages et les inconvénients du cloud computing.

I.2. Définitions

Un cloud est né comme un espace sur une infrastructure de réseau où les ressources informatiques telles que le matériel informatique, le stockage, les bases de données, les réseaux, systèmes d'évaluation, et même des applications logicielles entières sont disponibles instantanément [3], Le cloud computing est une extension du calcul parallèle, du calcul distribué et du calcul en grille. La virtualisation, la distribution et l'extensibilité dynamique sont les caractéristiques de base du cloud computing. [4].

“Cloud computing is a style of computing in which scalable and elastic IT-enabled capabilities are delivered as a service using Internet technologies.”

—Gartner Glossary [5]

Un Cloud est composée d'informatique distribuée, d'informatique en grille, d'utilitaires l'informatique et l'informatique autonome. Dans le cloud computing, les utilisateurs ne savent pas où les services sont situés dans quelle partie de l'infrastructure. Les utilisateurs utilisent uniquement le service via le paradigme de l'infrastructure cloud et payer pour les services demandés. L'infrastructure cloud fournit un accès à la demande à certaines ressources partagées et prestations de service. Ces services sont fournis en tant que service sur un réseau et sont accessibles Sur internet. [6]

Donc, en résumé : « Le cloud est une informatique parallèle et distribuée système consistant en une agrégation d'éléments interconnectés et ordinateurs virtualisés provisionnés dynamiquement et livré sous la forme d'une ou plusieurs ressources informatiques unifiées basées sur les accords de niveau de service (SLA) établis par négociation entre le prestataire de services et les clients » [7]

A partir les définitions précédentes, et avec la définition générale du NIST* le cloud est « un modèle pour permettre un accès réseau pratique et à la demande à un pool partagé de ressources informatiques configurables (par exemple, réseaux, serveurs, stockage, applications et services) qui peuvent être rapidement provisionnés et libérés avec un effort de gestion minimal ou une interaction avec le fournisseur de services ». [8]

Par rapport aux deux autres définitions, NIST fournit une définition relativement plus objective et spécifique, qui non seulement définit le concept de cloud dans son ensemble, mais

* NIST : Le National Institute of Standards and Technology, est une agence du département du Commerce des États-Unis. Son but est de promouvoir l'économie en développant des technologies (Voir : www.nist.gov)

spécifie également les caractéristiques essentielles du cloud computing et des modèles de livraison et de déploiement.

I.2.1. Terminologie « Cloud »

La première mention publique du Mot Cloud en 1994 dans un Article de l'auteur et journaliste Steven Levy. L'article parlait d'une société appelée General Magic. L'entreprise a été fondée par les créateurs de l'ordinateur Macintosh, Bill Atkinson et Andy Herzfeld En 1994 [5]:

” Now, instead of just having a device to program, we now have the entire Cloud out there, where a single program can go and travel to many different sources of information and create sort of a virtual service”.

—Bill and Andy’s Excellent Adventure II, *Wired* 1994

Les systèmes distribués à grande échelle ont gagné de l’attention depuis les années 90 en raison de la croissance d’internet, et la popularité des grilles de calcul et des Clouds composées de milliers d’ordinateurs. Cet ensemble de machines interconnectées permet d’offrir à ses utilisateurs une capacité de calcul importante [9].

Pourquoi l'industrie a-t-elle commencé à utiliser les termes Cloud ? et a-t-elle Data est dans le cloud si ce n'est pas le cas ? Avec des équipements de mise en réseau plutôt que dans un bâtiment séparé, les ingénieurs ont déclaré que les données le centre était dans le nuage et la terminologie saisie. [10].

I.3. Architecture d’un système cloud

Une architecture cloud est un modèle pour l'intégration des technologies qui permettent de créer des clouds, environnements informatiques qui dissocient, regroupent et partagent des ressources évolutives sur un réseau. L'architecture cloud indique la manière dont les composants et fonctionnalités nécessaires à la conception d'un cloud sont connectés afin de fournir une plateforme en ligne sur laquelle les applications s'exécuteront.

Les clouds sont considérés comme des PaaS (Platform-as-a-Service) puisque le fournisseur de cloud fournit à la fois la plateforme et l'infrastructure informatique sous-jacente. La conception de l'architecture d'une plateforme cloud implique plus que la séparation des capacités d'un ordinateur de ses composants matériels, ce que font les fournisseurs pour proposer une infrastructure cloud aux utilisateurs. Pour concevoir l'architecture, il faut ajouter des niveaux de développement supplémentaires pour intégrer la conteneurisation, l'orchestration, les API, le routage, la sécurité, la gestion et des logiciels d'automatisation. Il est

aussi important de ne pas négliger la conception de l'expérience utilisateur si vous souhaitez créer une expérience en ligne navigable. [11]

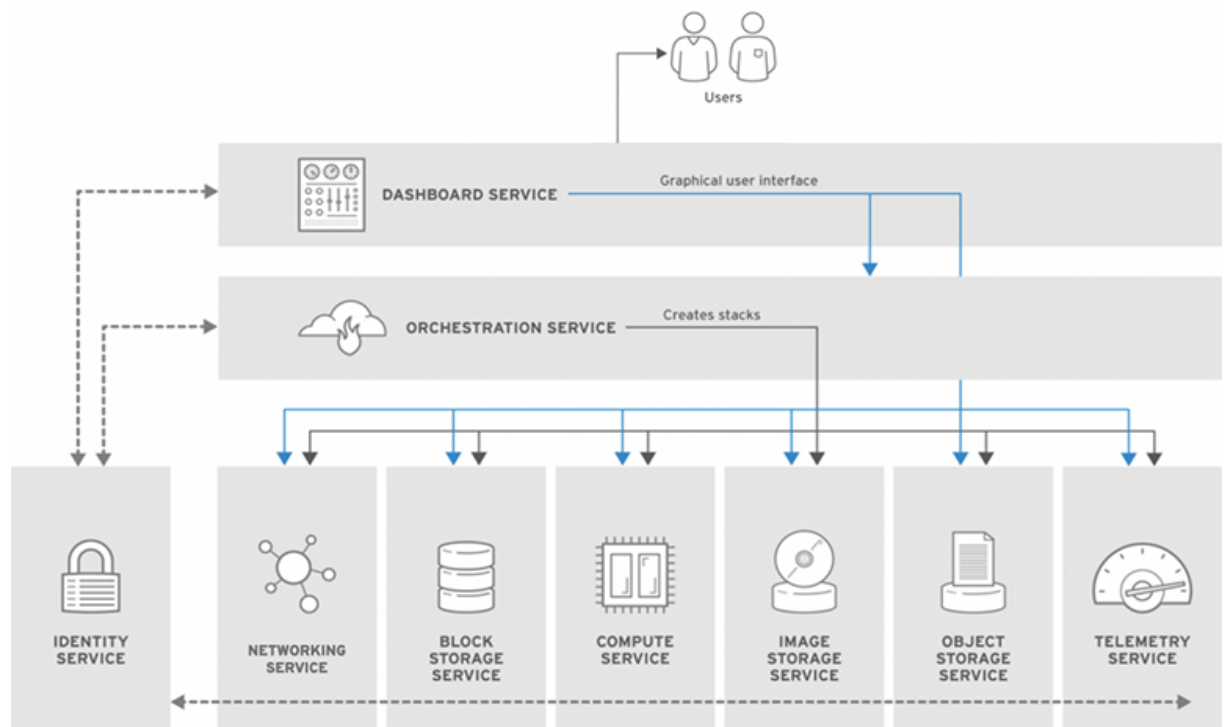


Fig I. 1: exemple d'architecture générale d'un cloud [12]

L'architecture cloud peut varier selon ce que vous essayez de faire, cependant, la plupart des clouds nécessitent du matériel, des solutions de middleware, des solutions de gestion et des logiciels d'automatisation. La plupart des clouds font appel à la virtualisation pour extraire les ressources matérielles et les transférer dans des data gérés de façon centralisée, alors que d'autres clouds, les « clouds nus », connectent directement les clients au matériel. [11]

I.4. Caractéristiques essentielles

Bien que certains critères puissent être soumis à débat, il y a selon moi certaines notions indiscutables qui caractérisent un service en Cloud [8]:

Libre-service à la demande : Un consommateur peut fournir unilatéralement des capacités informatiques, telles que le temps du serveur et le stockage réseau, selon les besoins, automatiquement, sans nécessiter d'interaction humaine avec chaque fournisseur de services.

Large accès au réseau : Les fonctionnalités sont disponibles sur le réseau et accessibles via des mécanismes standard qui favorisent l'utilisation par des plates-formes hétérogènes de clients légers ou lourds (par exemple, téléphones mobiles, tablettes, ordinateurs portables et stations de travail).

Mise en commun des ressources : Les ressources informatiques du fournisseur sont regroupées pour servir plusieurs consommateurs à l'aide d'un modèle multi-locataire, avec différentes ressources physiques et virtuelles affectées et réaffectées dynamiquement en fonction de la demande des consommateurs. Il existe un sentiment d'indépendance géographique dans la mesure où le client n'a généralement aucun contrôle ou connaissance de l'emplacement exact des ressources fournies, mais peut être en mesure de spécifier l'emplacement à un niveau d'abstraction plus élevé (par exemple, pays, état ou centre de données). Des exemples de ressources incluent le stockage, le traitement, la mémoire et la bande passante du réseau.

Élasticité rapide : Les capacités peuvent être provisionnées et libérées de manière élastique, dans certains cas automatiquement, pour évoluer rapidement vers l'extérieur et vers l'intérieur en fonction de la demande. Pour le consommateur, les capacités disponibles pour l'approvisionnement semblent souvent illimitées et peuvent être appropriées en n'importe quelle quantité à tout moment.

Prestation mesurée : Les systèmes cloud contrôlent et optimisent automatiquement l'utilisation des ressources en tirant parti d'une capacité de mesure à un certain niveau d'abstraction adapté au type de service (par exemple, stockage, traitement, bande passante et comptes d'utilisateurs actifs). L'utilisation des ressources peut être surveillée, contrôlée et signalée, offrant une transparence à la fois au fournisseur et au consommateur du service utilisé.

I.5. Les Services de Cloud Computing

Le cloud computing peut être divisé en trois classes selon leur niveau d'abstraction, à savoir (voire Fig I.1) [7]:

- A) Logiciel en tant que service (SAAS)
- B) Plate-forme en tant que service (PAAS)
- C) Infrastructure-as-a-service (IAAS)

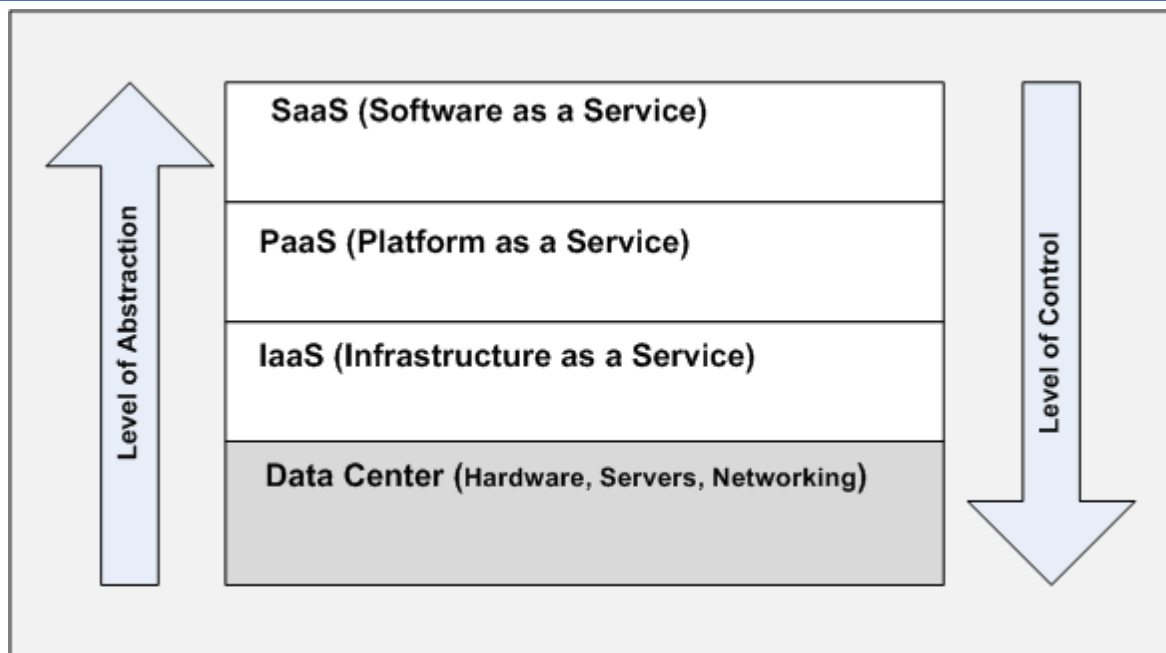


Fig I. 2: Cloud Framework (services) selon les niveaux d'abstract [7]

A) IaaS

Dans le cas le plus simple, une entreprise IaaS propose des ressources physiques, telles qu'un bâtiment, l'électricité et la climatisation. En règle générale, les entreprises IaaS fournissent également des serveurs, des équipements de mise en réseau et des installations de stockage de données de base (par exemple, le stockage de blocs sur disque). Une entreprise IaaS peut offrir à ses clients de nombreux services supplémentaires, tels que des équilibreurs de charge, la sauvegarde des données, la sécurité du réseau, un moyen de démarrer des serveurs physiques et virtualisés et l'attribution d'adresses Internet. Un client IaaS n'a pas besoin de gérer ou de contrôler l'infrastructure cloud. Un client peut choisir les systèmes d'exploitation et les applications qui s'exécutent, et peut avoir la possibilité de contrôler l'accès au réseau (par exemple, pour configurer un pare-feu). Les entreprises IaaS les plus avancées utilisent des systèmes d'exploitation qui peuvent faire évoluer les services du client et les installations allouées à un client vers le haut ou vers le bas en fonction des besoins. [10]

B) PaaS

L'objectif principal de PaaS est une installation qui permet à un client de créer et de déployer un logiciel dans un cloud sans consacrer d'efforts à la configuration ou à la gestion de l'installation sous-jacente. Une entreprise proposant le PaaS peut fournir à la fois une infrastructure de base et des installations pour le développement et le déploiement de logiciels. L'infrastructure de base comprend de nombreuses installations IaaS, telles que des serveurs, des installations de stockage, des systèmes d'exploitation, des bases de données et des connexions réseau. Les installations de développement et de déploiement de logiciels comprennent des

compilateurs, des intergiciels, des bibliothèques de programmes, des systèmes d'exécution (par exemple, Java Runtime et .NET Runtime), et les services qui hébergent les applications d'un client. Pour souligner son objectif de fournir un environnement pratique pour le développement de logiciels, PaaS est parfois appelé plateforme d'application en tant que service (aPaaS) et était auparavant appelé Framework en tant que service (FaaS), en référence aux cadres de programmation. [10]

C) SaaS

Le logiciel en tant que service fait référence à un modèle d'abonnement dans lequel un client paie des frais mensuels pour utiliser un logiciel plutôt que de faire un achat unique. Le cloud computing a permis à l'industrie SaaS en offrant aux fournisseurs SaaS un moyen d'adapter leurs offres pour gérer arbitrairement de nombreux clients. Lorsqu'un utilisateur accède à une application SaaS, l'application s'exécute sur un serveur dans un centre de données cloud plutôt que sur l'ordinateur de l'utilisateur. Les fichiers créés par l'utilisateur sont stockés dans le centre de données cloud plutôt que sur l'appareil local de l'utilisateur. Les services SaaS bien connus incluent Office365 de Microsoft dans lequel chaque client paie des frais mensuels pour utiliser les programmes de la suite Office, tels que Word, Excel et PowerPoint. [10]

D) Cas spécial : (DaaS)

De nombreux groupes ont adopté l'expression en tant que service pour décrire leur segment de marché particulier, y compris le réseau en tant que service, la sécurité en tant que service, la reprise après sinistre en tant que service et le backend mobile en tant que service (fournissant une communication entre les applications mobiles et le logiciel cloud qu'ils utilisent). Une forme particulière de SaaS se distingue comme particulièrement pertinente pour le cloud computing. Connu sous le nom de Desktop as a Service (DaaS), le système implémente l'accès au bureau à distance. Comme pour les autres offres SaaS, un utilisateur exécute une application sur un appareil local (c'est-à-dire athinclient) qui connecte l'utilisateur au logiciel DaaS exécuté dans un centre de données cloud. Au lieu de fournir un accès comme dans l'application, cependant, DaaS peint un bureau sur l'écran de l'utilisateur et permet à l'utilisateur de cliquer sur des icônes, d'exécuter des applications, de parcourir des fichiers et d'effectuer les actions exactement comme si le bureau était local. Le bureau que l'utilisateur voit, le système d'exploitation qui fournit le bureau et les applications qu'un utilisateur appelle s'exécutent tous sur un serveur dans le cloud au lieu de l'appareil local de l'utilisateur. [10]

I.6. Cloud technologies et infrastructure

Les mécanismes d'infrastructure cloud sont des blocs de construction fondamentaux des environnements cloud qui établissent des artefacts primaires pour former la base de l'architecture technologique cloud fondamentale.

La relation entre l'infrastructure et l'architecture peut expliquer par cet exemple ; Imaginez que vous construisez une maison : *l'infrastructure cloud correspond aux matériaux et l'architecture cloud au plan.*

Les mécanismes d'infrastructure cloud suivants sont décrits : [13]

I.6.1. Périmètre de réseau logique (Logical Network Perimeter)

Défini comme l'isolement d'un environnement réseau du reste d'un réseau de communication, le périmètre de réseau logique établit une limite de réseau virtuel qui peut englober et isoler un groupe de ressources informatiques liées basées sur le cloud qui peuvent être physiquement distribuées.

Ce mécanisme peut être mis en œuvre pour :

- isoler les ressources informatiques dans un cloud des utilisateurs non autorisés
- isoler les ressources informatiques dans un cloud des non-utilisateurs
- isoler les ressources informatiques dans un cloud des consommateurs du cloud
- contrôler la bande passante disponible pour les ressources informatiques isolées

Les périmètres de réseau logiques sont généralement établis via des périphériques réseau qui alimentent et contrôlent la connectivité d'un centre de données et sont généralement déployés en tant qu'environnements informatiques virtualisés qui incluent :

- Pare-feu virtuel - Une ressource informatique qui filtre activement le trafic réseau vers et depuis le réseau isolé tout en maîtrisant ses interactions avec Internet.
- Réseau virtuel - Généralement acquis via des VLAN, cette ressource informatique isole l'environnement réseau au sein de l'infrastructure du centre de données.

I.6.2. Serveur virtuel

Un serveur virtuel est une forme de logiciel de virtualisation qui émule un serveur physique. Les serveurs virtuels sont utilisés par les fournisseurs de cloud pour partager le même serveur physique avec plusieurs consommateurs de cloud en fournissant aux consommateurs de cloud des instances de serveur virtuel individuelles. Le nombre d'instances qu'un serveur physique donné peut partager est limité par sa capacité.

En tant que mécanisme de base, le serveur virtuel représente le bloc de construction le plus fondamental des environnements cloud. Chaque serveur virtuel peut héberger de nombreuses ressources informatiques, des solutions basées sur le cloud et divers autres mécanismes de cloud computing. L'instanciation de serveurs virtuels à partir de fichiers image est un processus d'allocation de ressources qui peut être effectué rapidement et à la demande.

I.6.3. Périphérique de stockage en nuage

Le mécanisme de périphérique de stockage cloud représente les périphériques de stockage conçus spécifiquement pour le provisionnement basé sur le cloud. Les instances de ces périphériques peuvent être virtualisées, de la même manière que les serveurs physiques peuvent générer des images de serveur virtuel. Les dispositifs de stockage en nuage sont généralement capables de fournir une allocation de capacité à incréments fixes à l'appui du mécanisme de paiement à l'utilisation. Les périphériques de stockage en nuage peuvent être exposés pour un accès à distance via des services de stockage en nuage.

Les mécanismes de périphérique de stockage cloud fournissent des unités logiques communes de stockage de données, telles que :

- Fichiers – Les collections de données sont regroupées dans des fichiers situés dans des dossiers.
- Blocs - Le niveau de stockage le plus bas et le plus proche du matériel, un bloc est la plus petite unité de données qui est toujours accessible individuellement.
- Ensembles de données – Les ensembles de données sont organisés dans un format tabulaire, délimité ou d'enregistrement.
- Objets – Les données et leurs métadonnées associées sont organisées sous forme de ressources Web.

I.6.4. Moniteur d'utilisation du cloud

Le mécanisme de surveillance de l'utilisation du cloud est un logiciel léger et autonome chargé de collecter et de traiter les données d'utilisation des ressources informatiques.

Selon le type de mesures d'utilisation qu'ils sont conçus pour collecter et la manière dont les données d'utilisation doivent être collectées, les moniteurs d'utilisation du cloud peuvent exister dans différents formats. Les prochaines sections décrivent trois formats de mise en œuvre courants basés sur des agents. Chacun peut être conçu pour transmettre les données d'utilisation collectées à une base de données de journaux à des fins de post-traitement et de création de rapports.

I.6.5. Réplication des ressources

Définie comme la création de plusieurs instances de la même ressource informatique, la réplication est généralement effectuée lorsque la disponibilité et les performances d'une ressource informatique doivent être améliorées. La technologie de virtualisation est utilisée pour mettre en œuvre le mécanisme de réplication des ressources afin de répliquer les ressources informatiques basées sur le cloud.

I.6.6. Environnement prêt à l'emploi (Ready-Made Environment)

Le mécanisme d'environnement prêt à l'emploi est un composant déterminant du modèle de livraison cloud PaaS qui représente une plate-forme cloud prédéfinie composée d'un ensemble de ressources informatiques déjà installées, prêtes à être utilisées et personnalisées par un consommateur cloud. Ces environnements sont utilisés par les consommateurs de cloud pour développer et déployer à distance leurs propres services et applications dans un cloud. Les environnements prêts à l'emploi typiques incluent des ressources informatiques préinstallées, telles que des bases de données, des intergiciels, des outils de développement et des outils de gouvernance.

I.7. Modèles de déploiement

Les clouds sont déployés de différentes manières, selon les périmètres d'utilisation. Il existe quatre principaux modèles de déploiement cloud. [14]

- *Le cloud public* est le paradigme standard du cloud computing, dans lequel un fournisseur de services met des ressources, telles que des applications et du stockage, à la disposition du grand public via Internet. Les fournisseurs de services facturent sur une base informatique fine. Des exemples de clouds publics incluent Amazon Elastic Compute Cloud (EC2), Blue Cloud d'IBM, Sun Cloud, Google AppEngine et Windows Azure Services Platform.

- *Le cloud privé* ressemble plus à un concept marketing qu'au sens traditionnel. Il décrit une architecture informatique propriétaire qui fournit des services à un nombre limité de personnes sur des réseaux internes. Les organisations qui ont besoin d'un contrôle précis sur leurs données préféreront le cloud privé, afin qu'elles puissent bénéficier de tous les avantages d'évolutivité, de mesure et d'agilité d'un cloud public sans céder le contrôle, la sécurité et les coûts récurrents à un fournisseur de services. eBay et HP CloudStart génèrent des déploiements de cloud privé.

- *Le cloud hybride* utilise une combinaison de cloud public, de cloud privé et même d'infrastructures locales, ce qui est typique pour la plupart des fournisseurs informatiques. La

stratégie hybride consiste à placer correctement les charges de travail en fonction des coûts, des facteurs opérationnels et de conformité. Les principaux fournisseurs, dont HP, IBM, Oracle et VMware, créent des plans appropriés pour tirer parti d'un environnement mixte, dans le but de fournir des services à l'entreprise. Les utilisateurs peuvent déployer une application hébergée sur une infrastructure hybride, dans laquelle certains nœuds s'exécutent sur du matériel physique réel et d'autres sur des instances de serveur cloud.

I.8. Challenges de recherche en environnement de cloud computing

Malgré le succès initial et la popularité des clouds et la grande disponibilité des fournisseurs et des outils, un nombre important de défis (challenges) et de risques sont inhérents à ce nouveau modèle d'informatique. Les fournisseurs, les développeurs et les utilisateurs finaux doivent tenir compte de ces défis et risques pour tirer pleinement parti du cloud computing. Les problèmes à résoudre incluent la confidentialité des utilisateurs, les données... etc. Par suit on a souligné quelque facteurs critique [15] :

II.8.1. Sécurité, confidentialité et confiance

La sécurité et la confidentialité affectent l'ensemble de la pile de cloud computing, car il y a une utilisation massive de services et d'infrastructures tiers qui sont utilisés pour héberger des données importantes ou pour effectuer des opérations critiques. Dans ce scénario, la confiance envers les fournisseurs est fondamentale pour garantir le niveau de confidentialité souhaité pour les applications hébergées dans le cloud.

II.8.2 Verrouillage des données et standardisation

L'une des principales préoccupations des utilisateurs du cloud computing est de voir leurs données verrouillées par un certain fournisseur. Les utilisateurs peuvent souhaiter déplacer des données et des applications d'un fournisseur qui ne répond pas à leurs besoins.

Cependant, dans leur forme actuelle, les infrastructures et plates-formes de cloud computing n'utilisent pas de méthodes standard de stockage des données et des applications des utilisateurs. Par conséquent, ils n'interagissent pas et les données des utilisateurs ne sont pas portables.

II.8.3. Disponibilité, tolérance aux pannes et reprise après des échecs

On s'attend à ce que les utilisateurs aient certaines attentes quant au niveau de service à fournir une fois leurs applications déplacées vers le cloud. Ces attentes incluent la disponibilité du service, ses performances globales et les mesures à prendre en cas de problème dans le

système ou ses composants. En résumé, les utilisateurs recherchent une garantie avant de pouvoir déplacer confortablement leur entreprise vers le cloud.

II.8.4. Gestion des ressources et efficacité énergétique

Un défi important auquel sont confrontés les fournisseurs de services de cloud computing est la gestion efficace des pools de ressources virtualisées. Les ressources physiques telles que les cœurs de processeur, l'espace disque et la bande passante du réseau doivent être découpées en tranches et partagées entre les machines virtuelles exécutant des charges de travail potentiellement hétérogènes.

Un autre défi concerne la quantité exceptionnelle de données à gérer dans diverses activités de gestion de VM. Une telle quantité de données est le résultat de capacités particulières des machines virtuelles, y compris la capacité de voyager dans l'espace (c'est-à-dire la migration) et le temps (c'est-à-dire les points de contrôle).

I.9. Avantages et désavantage de Cloud computing

I.9.1. Avantages du Cloud computing

Le cloud computing est en pleine croissance. Flexibilité, réduction des coûts, optimisation de l'accessibilité aux outils, support simple et performant du travail collaboratif, il offre de nouveaux avantages aux entreprises et aux utilisateurs comme la suite : [16]

A. Avantages au niveau de la stratégie

Au niveau de la stratégie, de nombreuses entreprises s'appuient sur le Cloud pour alimenter de nouvelles stratégies commerciales et chercher des sources concurrentielles. L'optimisation des ressources et les économies d'échelle augmentent en théorie les marges. L'impact du Cloud sur la stratégie se manifeste notamment par la création de nouveaux <business models >, qui affectent tout l'écosystème de l'entreprise. L'enjeu consiste à disposer des bonnes informations au bon moment pour prendre les bonnes décisions. Cela passe par la mise en relation, le partage et la combinaison de l'ensemble des actifs stratégiques de l'organisation. Pour l'heure, les DSI sont limitées par les capacités techniques des solutions de Cloud, notamment en termes de sécurité des informations et de portabilité des données.

B. Avantages au niveau des fonctions et des processus métier

Au niveau des processus et des fonctions métier, les entreprises cherchent avant tout la performance, le partage des ressources (afin d'accéder à des services auxquels elles ne pouvaient pas prétendre auparavant), une collaboration plus étroite, davantage d'intégration, ainsi qu'une meilleure coordination interprocessus. Or les solutions de Cloud computing favorisent la

coordination des processus et des fonctions du métier. D'ailleurs, certains des plus grands succès du Cloud computing concernent à ce jour des solutions de collaboration, qui permettent aux groupes et aux communautés de travailler ensemble de manière innovante.

C. Avantages opérationnels

Les principaux avantages opérationnels qu'offre une solution de Cloud computing concernent la baisse des coûts de production des services informatiques, grâce à la disponibilité et l'élasticité des ressources informatiques, ainsi qu'à des systèmes de facturation portant sur la consommation réelle de services, par opposition aux

Systemes de forfaits (pour lequel le client paie, même s'il ne consomme rien). Les DSI estiment qu'ils pourraient réaliser des économies de 10 à 50 % sur ces coûts de production. Il s'agit en outre de commercialiser plus rapidement de nouvelles applications et d'accélérer leur mise à jour.

I.9.2. Inconvénients du Cloud computing

Selon le géant INTEL, qui est déjà faire nombreuses recherches sur les clouds et les platforms cloud scientifique, déclarés deux raisons passives concernant les clouds computing ; ces raisons sont [12] :

- **Confidentialité et sécurité des données** : les données sont hébergées en dehors de l'entreprise. Le fournisseur proposant le service héberge les données de l'entreprise utilisatrice. Cela peut donc poser un risque potentiel pour l'entreprise de voir ses données mal utilisées ou volées. Il s'agit donc de s'assurer que le fournisseur dispose d'une sécurité suffisante et qu'il propose une politique de confidentialité concernant les données de l'utilisateur.
- **Dépendance** : si l'entreprise souhaite des fonctionnalités très spécifiques, il peut être difficile de convaincre le fournisseur de proposer ces fonctionnalités. Et en général, s'il y a un problème, l'entreprise est tributaire du service client du fournisseur. Il s'agit donc de choisir un fournisseur en qui l'on a confiance

I.10. Clouds Exemples

Dans le monde technologique, certains grands établissements dominant le marché cloud avec service sécurisés et fiables. [17]

I.10.1. Google (Google AppEngin)

Le Giant Google est certainement l'une des plus grandes entreprises commerciales de Google, et ils offrent quelques outils pour aider à dessiner clients à leur cloud. Dans cette section, on peut citer quelque avantages les suivants :

- Écrivez le code une seule fois et déployez-le L'approvisionnement et la configuration de plusieurs machines pour le service Web et le stockage des données peuvent s'avérer coûteux et chronophages. Google App Engine facilite le déploiement d'applications Web en fournissant dynamiquement les ressources informatiques nécessaires. Les développeurs écrivent le code et Google App Engine s'occupe du reste.

- Absorber les pics de trafic Lorsqu'une application Web gagne en popularité, l'augmentation soudaine du trafic peut être écrasante pour les applications de toutes tailles, des startups aux grandes entreprises qui doivent réorganiser leurs bases de données et leurs systèmes entiers plusieurs fois par an. Grâce à la réplication et à l'équilibrage de charge automatiques, Google App Engine facilite le passage d'un utilisateur à un million en tirant parti de Big_table et d'autres composants de l'infrastructure évolutive de Google.

- Intégration facile avec d'autres services Google Il est inutile et inefficace pour les développeurs d'écrire des composants tels que l'authentification et l'e-mail à partir de zéro pour chaque nouvelle application. Les développeurs utilisant Google App Engine peuvent utiliser des composants intégrés et la bibliothèque d'API plus large de Google qui fournissent des fonctionnalités plug-and-play (PnP) pour des fonctionnalités simples mais importantes.

I.10.2. EMC

EMC Corporation est le leader mondial des produits, services et solutions de stockage et de gestion des informations qui aident les entreprises à tirer parti de leurs informations.

Ils ont leurs doigts dans toutes sortes de secteurs de l'informatique en nuage et de la virtualisation. Par exemple, au début de 2009, EMC a fait monter les enchères en pariant que la virtualisation allait être la « prochaine nouveauté » dans le monde de l'informatique. Il a présenté son V-Max symétrique

Systeme en avril 2009, affirmant qu'il s'agit du premier système de gestion à prendre en charge les centres de données virtuels haut de gamme.

Le système permet aux clients ayant de vastes besoins de stockage de gérer et d'étendre facilement les systèmes de stockage sans interférer avec les opérations quotidiennes. Ce système permet à plusieurs centres de données d'être exécutés comme s'ils n'en étaient qu'un, ce qui facilite grandement leur gestion et plus efficace.

Les technologies

EMC va bien au-delà de la gestion des centres de données virtualisés. Leurs autres domaines de l'expertise comprennent :

- **Archivage** : Création d'archives en ligne accessibles qui offrent un coût opérationnel réduit en réduisant les fenêtres de sauvegarde et en accélérant les restaurations.
- **Sauvegarde et restauration** : Différents outils combinent les offres de gestion de la restauration, les technologies de sauvegarde et les stratégies de gestion d'EMC pour garantir que vous disposez d'une solide pratique de sauvegarde et de restauration.
- **Gestion de contenu d'entreprise** : Les solutions basées sur le contenu contribuent à atténuer les risques sans imposer de technologies trop complexes à votre organisation.
- **Gestion intelligente de l'information** : L'utilisation de diverses technologies permet aux organisations de découvrir, de stocker et d'agir sur l'information de manière intelligente.
- **Gestion informatique** : La gestion informatique est simplifiée et son coût réduit grâce à l'automatisation, la virtualisation et l'efficacité des processus.
- **Réplication** : Les technologies de protection des données et de réplication à distance offrent des options de reprise après sinistre.
- Les organisations de sécurité peuvent déployer des produits dotés de fonctionnalités de contrôle d'accès, de protection des données et d'audit.

I.10.3. Microsoft

Microsoft propose un certain nombre de services cloud pour les organisations de toutes tailles, des entreprises aux boutiques familiales ou aux particuliers. Une bonne partie des offres cloud de Microsoft sont des variantes cloud de produits que les gens utilisent déjà, donc des versions cloud ne sont pas si difficiles à utiliser. Et le service fameux de Microsoft est Windows AZUR.

Windows Azure

Windows Azure est un système d'exploitation basé sur le cloud qui permet l'environnement de développement, d'hébergement et de gestion des services pour la plateforme de services Azure. Windows Azure offre aux développeurs un environnement de calcul et de stockage à la demande qu'ils peuvent utiliser pour héberger, mettre à l'échelle et gérer des applications Web via des centres de données Microsoft.

Pour créer des applications et des services, les développeurs peuvent utiliser les compétences Visual Studio qu'ils possèdent déjà. De plus, Azure prend en charge les normes existantes telles que SOAP, REST et XML.

Windows Azure peut être utilisé pour :

- Ajouter des fonctionnalités de service Web aux applications existantes.
- Créez et modifiez des applications, puis déplacez-les sur le Web.
- Créez, testez, déboguez et distribuez des services Web de manière efficace et peu coûteuse.
- Réduire les coûts de gestion informatique.

I.10. Conclusion

Dans ce chapitre nous avons introduit et défini la notion de cloud computing, tout en donnant une vision générale autour de ce concept. L'ordonnancement est un enjeu important qui influence considérablement les performances de l'environnement de cloud computing. Ce problème est d'autant plus difficile lorsqu'il y a plusieurs facteurs à prendre en compte, donc c'est un problème d'optimisation combinatoire, ou il est possible de trouver la solution optimale en utilisant des algorithmes. A cet égard, le chapitre suivant sera consacré à ce problème.

Chapitre II :
l'ordonnancement dans les Clouds Computing

II.1. Introduction

Après avoir les clouds dans le chapitre I ; le chapitre II par suite est traité l'ordonnancement dans les cloud comme un concept très importante pour mesurée l'efficacité d'une cloud service.

On a présenté dans ce chapitre la stratégie de l'ordonnancement et la planification et ça propres définitions ; caractéristique et concepts fondamentaux. Ainsi que les types des algorithmes associer à l'ordonnancement.

Du même, nous présentons les problèmes liés à l'ordonnancement dans pleueur niveaux (complexité des algorithmes de l'ordonnancement ; l'efficacité des algorithmes classique ; ...) et en fin nous avons en bref présenter un exemple de stratégie de l'ordonnancement créé par Amazon.

Notre étude porte également sur la logique floue, comme une solution aux problèmes d'ordonnancement dans les Clouds. nous avons présenté le concept de la logique floue, puis le lien entre celle-ci et les algorithmes d'ordonnancements dans le cloud.

II.2. Présentation des algorithmes d'ordonnancement des tâches

II.2.1. Définitions

Les algorithmes[†] de planification des tâches ou d'ordonnancement sont définis comme le mécanisme utilisé pour sélectionner les ressources pour exécuter des tâches afin d'obtenir moins de temps d'attente et d'exécution. [18]

Encore ; Les algorithmes d'ordonnancement des tâches sont définis comme un ensemble de règles et de politiques utilisées pour affecter des tâches aux ressources appropriées (CPU, mémoire et bande passante) pour obtenir le plus haut niveau possible de performance et d'utilisation des ressources. [19]

II.2.2. Notions liées à l'ordonnancement

Les technologies clés de la planification des ressources comprennent [20]:

- **Stratégies d'ordonnancement** : c'est le niveau supérieur de la gestion de la planification des ressources, qui nécessite à définir par les propriétaires et gestionnaires de centres de données. Il détermine principalement la ressource objective de planification et s'assure de satisfaire toutes les stratégies de manutention immédiatement requises lorsque les ressources sont insuffisantes.

- **Objectifs d'optimisation** : le centre de planification doit identifier différentes fonctions objectives pour déterminer les avantages et les inconvénients des différents types de planification. Maintenant, il y a un objectif optimal fonctions, telles que les coûts minimaux, les profits maximaux et l'utilisation maximale des ressources.

- **Algorithmes d'ordonnancement** : de bons algorithmes de planification doivent produire des résultats optimaux selon des fonctions objectives en un temps très court sans consommer trop Ressources. De manière générale, les problèmes que les algorithmes d'ordonnancement doivent résoudre sont essentiellement Les problèmes NP-difficiles, qui nécessitent une grande quantité de calculs et ne sont généralement pas utilisé. L'industrie utilise généralement des algorithmes d'ordonnancement optimaux approximatifs et utilise différents algorithmes d'ordonnancement pour différentes applications.

[†] - Un algorithme est un ensemble d'instructions simples permettant de trouver une solution à un problème. Il contient trois parties : entrée, méthode, sortie. L'entrée est un ensemble de paramètres à traiter. Le procédé comprend des procédures descriptibles, contrôlables et reproductibles pour réaliser l'objectif à l'aide de paramètres d'entrée. La sortie est le résultat du problème. (Voir : [18]).

- **Architecture du système d'ordonnancement** : elle est étroitement liée à l'infrastructure de base des données centres. De nos jours, la structure du système distribué à plusieurs étages illustrée à la fig II.1 est principalement utilisé.

- **Ordonnancement des ressources**

La planification des ressources est un processus par lequel les fournisseurs de ressources allouent des ressources à utilisateurs. Pour un centre de données Cloud, les types de stratégies de planification des ressources doivent envisager d'inclure les serveurs physiques, les clusters de serveurs physiques, les mémoires partagées, les bandes passantes, des images de machines virtuelles et des groupes de sécurité.

II.3. Système de planification de tâches dans le cloud computing

Le système d'ordonnancement des tâches dans le cloud computing passe par trois niveaux [7].

- *Le premier niveau de tâche* : est un ensemble de tâches (Cloudlets) qui est envoyé par les utilisateurs du cloud, qui sont nécessaires à l'exécution.

- *Le deuxième niveau de planification* : est responsable de la mise en correspondance des tâches avec ressources pour obtenir une utilisation maximale des ressources avec un minimum d'envergure. Le makepan est le temps d'exécution global pour toutes les tâches du début à la fin.

- *Le troisième niveau de VM* : est un ensemble de (VM) qui sont utilisées pour exécuter les tâches.

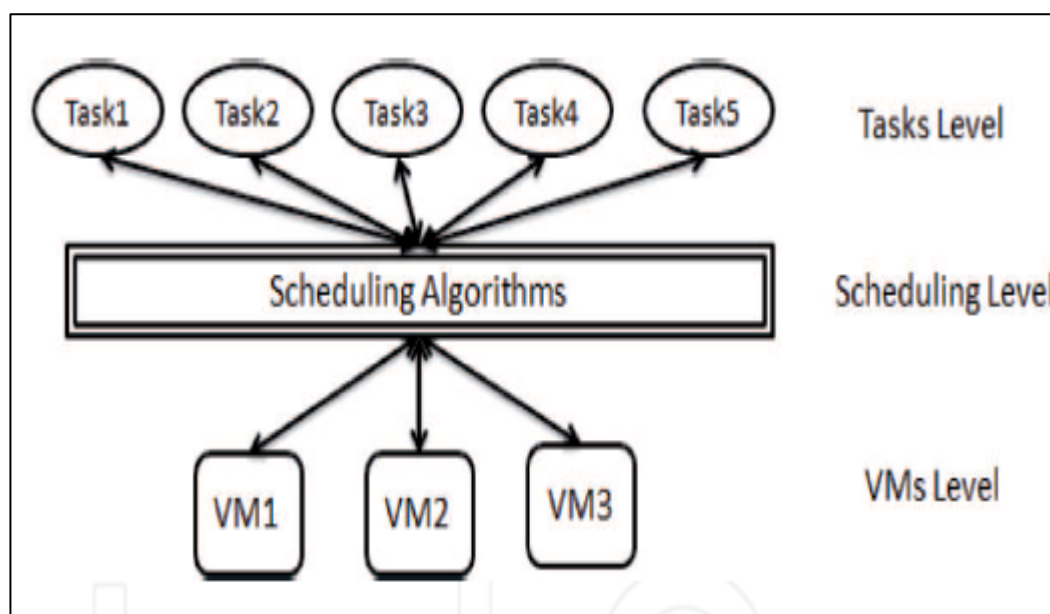


Fig II. 1: Schéma d'ordonnancement simple [1]

II.3.1. Niveaux d'ordonnancement

Dans l'environnement de cloud computing, il existe deux niveaux de planification algorithmes [1] :

- *Premier niveau* : au niveau de l'hôte où un ensemble de politiques pour distribuer les machines virtuelles dans l'hôte.
- *Deuxième niveau* : au niveau de la VM où un ensemble de politiques pour distribuer les tâches à la VM.

II.3.2. Avantages des algorithmes de planification de tâches

- Gérer les performances et la qualité de service du cloud computing.
- Gérer la mémoire et le CPU.
- Les bons algorithmes de planification maximisant l'utilisation des ressources tout en minimisant le temps total d'exécution de la tâche.
- Améliorer l'équité pour toutes les tâches.
- Augmenter le nombre de tâches accomplies avec succès.
- Planification des tâches sur un système en temps réel.
- Atteindre un débit système élevé.
- Amélioration de l'équilibre de charge.

II.4. Classifications des algorithmes d'ordonnancement des tâches

Les algorithmes de planification des tâches peuvent être classés comme suit : [21]

- **Ordonnancement immédiat** : lorsque de nouvelles tâches arrivent, elles sont planifiées sur les machines virtuelles directement.
- **Ordonnancement par lots** : les tâches sont regroupées dans un lot avant d'être envoyées ; ce genre est également appelé événements de mappage.
- **Ordonnancement statique** : est considéré comme très simple par rapport à l'ordonnancement dynamique ; il est basé sur une information préalable de l'état global du système. Cela fait ne prend pas en compte l'état actuel des machines virtuelles, puis divise tout le trafic de manière équivalente entre toutes les machines virtuelles de la même manière, comme le tourniquet (RR) et algorithmes d'ordonnancement aléatoire.

- **Ordonnancement dynamique** : prend en compte l'état actuel des VM et n'exige des informations préalables sur l'état global du système et diffuse les tâches en fonction de la capacité de toutes les machines virtuelles disponibles.

- **Ordonnancement préemptif** : chaque tâche est interrompue en cours d'exécution et peut être déplacé vers une autre ressource pour terminer l'exécution.

- **Ordonnancement non préemptive** : les VM ne sont pas réaffectées à de nouvelles tâches avant la fine exécution de la tâche planifiée.

Dans le tableau suivant, nous résumerons les algorithmes d'ordonnements.

Tableau II. 1: Analyse comparative des principales stratégies de planification [20] [21]

Type	Algorithmes	Stratégie	Objectif d'optimisation	Complexité	Points fort	Points faibles
Performance D'abord	First come, first served	Load balance	Rend l'utilisation d'une ressource atteindre la valeur moyenne	Bas	Prend en charge les performances de base des ressources (Les performances peuvent être configurées— domaine utilitaire de bonnes performances)	Assure principalement les performances tandis que d'autres caractéristiques peuvent ne pas être atteintes facilement
	Load balance	Haut Fiabilité	Fiabilise toutes les ressources atteindre l'exigence prédéfinie (comme assurer le service, cela stratégie peut être utilisée à 99,95 % le temps)	Bas	Assure la fiabilité du service	Si différentes exigences existent, beaucoup de ressources et de coûts sont ajoutés
	Improve reliability	Maximum Demande de l'utilisateur la satisfaction	Pas de fonction d'objectif explicite	Bas	Classifie le niveau d'utilisateur selon la priorité, le niveau de sécurité, etc.	Ne peut être alloué qualitativement que s'il n'est pas alloué quantitativement
Coût D'abord	Improve overall utilization	Maximum utility	Optimisation de l'utilisation des ressources = toutes les ressources informatiques du centre de données sont utilisées au maximum	Plus haute	Augmente l'utilisation des ressources à moindre coût	Ne peut pas satisfaire d'autres exigences de performances relatives, telles que l'équilibrage de charge, la fiabilité et un service rapide

	Maximum profit	Maximum profits	Bénéfices maximum = Max (revenu de tous types de ressources ; Coût de utilisant tous les types de ressources)	Haut	S'adapte aux exigences de profit de la plupart des fournisseurs d'affaires	Ne peut pas satisfaire d'autres exigences de performances relatives telles que l'équilibrage de charge, la fiabilité et un service rapide
	Minimum operation costs	Coût minimal	Coût minimum = Minimum (coût de toutes les ressources)	Haut	S'adapte aux exigences de profit de la plupart des fournisseurs	Ne peut pas satisfaire d'autres exigences de performances relatives telles que l'équilibrage de charge, la fiabilité et un service rapide

II.5. Combiner les stratégies d'ordonnancement (Hybridation des algorithmes)

Avant tout, la sélection de différentes stratégies doit tenir compte à la fois des exigences de service et les objectifs commerciaux dans la pratique. Dans le tableau II.2, la compatibilité de chaque ordonnancement de base stratégie est collectée. Si le service interne d'une entreprise est la première exigence, alors le coût minimum, l'utilité et l'équilibre de la charge doivent être pris en compte. Si application métier est la principale exigence, un profit maximum doit être envisagé. Par exemple, la stratégie de profit maximum doit être sélectionnée et configurée avec les équilibres de charge et stratégies de fiabilité comme contraintes.

Étant donné qu'une stratégie combinée peut entraîner de nombreuses nouvelles stratégies d'ordonnancement, *les 7 les stratégies d'ordonnancement essentielles peuvent créer 128 stratégies combinées*. Un recommandé méthode consisterait à définir une stratégie de planification de base comme stratégie principale, puis définir d'autres stratégies comme contraintes pour générer de nouvelles stratégies. Ici le maximum la compatibilité d'utilisation avec une haute fiabilité est utilisée comme exemple :

L'utilisation maximale signifie que l'utilisation de tous les serveurs physiques actifs atteint le niveau maximum (il y a une limite supérieure pour assurer la performance du service), donc certains serveurs physiques seraient arrêtés si nécessaire. Si une haute fiabilité est également considérée (comme dans la copie maîtresse), une contrainte supplémentaire qui pourrait influencer la copie maîtresse une copie peut être ajoutée en fonction d'autres contraintes avant d'arrêter les serveurs physiques. Si cette contrainte est déclenchée, l'opération d'arrêt ne se produise pas. De la même manière, lors de l'allocation de machines virtuelles à un serveur physique pour améliorer l'utilisation des ressources, satisfaire une haute fiabilité pourrait également être ajouté comme une contrainte.

Tableau II. 2 : Compatibilité (1) ou incompatibilité (0) des relations de la stratégie de planification intégrée de base [20]

	Satisfaction maximale des demandes des utilisateurs	Load balance	Haut fiabilité	Maximum utilisation	Maximum profit	Minimum Coût	Énergie Efficacité	Optimisation des capacités
Satisfaction maximale des demandes des utilisateurs	/	1	1	0	0	0	1	1
Load balance		/	1	0	0	0	1	1
Haut fiabilité			/	0	1	0	1	1
Maximum utilisation				/		1	0	0
Maximum profit					/		0	0
Minimum Coût						/		1
Énergie Efficacité							/	
Optimisation des capacités								/

II.6. Problèmes, algorithmes et complexité de l'ordonnancement

Le problème d'ordonnancement est le problème de faire correspondre des éléments de différents ensembles, qui est formellement exprimé par un triplet (E, S, O) , où [22] [14]:

- E est l'ensemble des exemples, dont chacun est une instance de problème.
- S est l'ensemble des solutions réalisables pour l'exemple.
- O est l'objet du problème.

Le problème d'ordonnancement peut être classé en deux catégories en fonction de l'objet O : problème d'optimisation et problème de décision. Un problème d'optimisation nécessite de trouver la meilleure solution parmi toutes les solutions réalisables dans l'ensemble S. Différent de l'optimisation, le but du problème de décision est relativement facile. Pour une solution réalisable spécifiée $s \in S$, le problème a besoin d'une réponse positive ou négative pour savoir si l'objet O est atteint. De toute évidence, le problème d'optimisation est plus difficile que le problème de décision, car la solution spécifiée ne se compare qu'à une solution de seuil dans le problème de décision, au lieu de toutes les solutions réalisables dans le problème d'optimisation.

D'une manière générale, il n'existe pas d'algorithme d'ordonnancement absolument parfait, car les objectifs d'ordonnancement peuvent entrer en conflit les uns avec les autres. Un bon ordonnanceur implémente un compromis approprié, ou applique une combinaison d'algorithmes d'ordonnancement selon différentes applications. Un problème peut être résolu en quelques secondes, heures ou même années selon l'algorithme appliqué. L'efficacité d'un algorithme est évaluée par le temps nécessaire à son exécution. Le temps d'exécution d'un algorithme est défini comme une fonction de complexité temporelle reliant la longueur d'entrée au nombre d'étapes. [14]

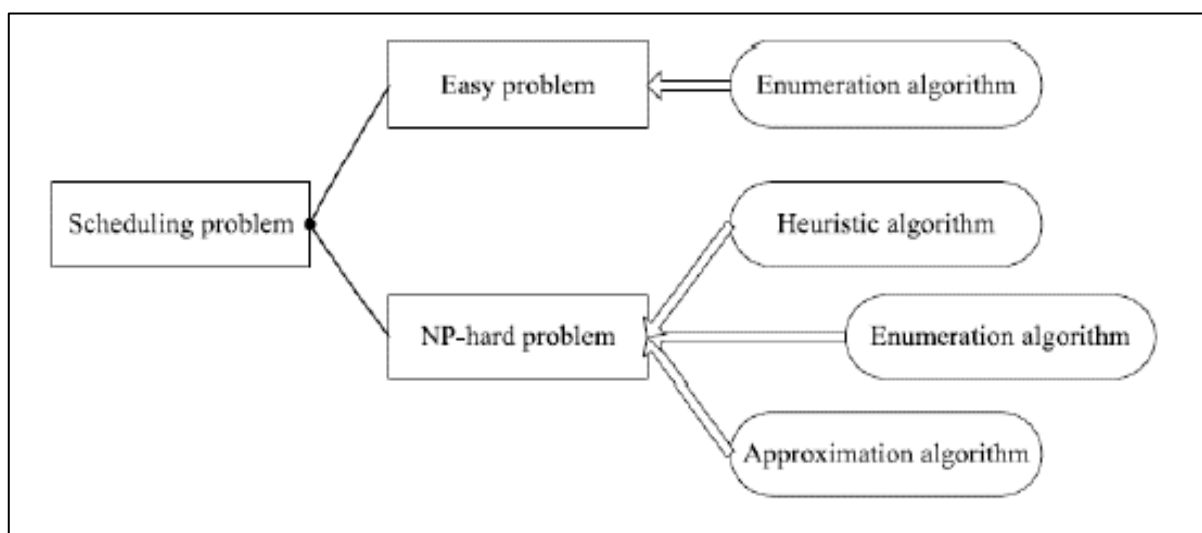


Fig II. 2: Vu schématique de problème d'ordonnancement [14]

II.7. Contraintes des stratégies d'ordonnancement [20]

- A- Espace :** (association et anti-association) La règle d'association consiste à placer deux machines virtuelles ou plus sur le même serveur. L'anti-association signifie que deux ou plusieurs machines virtuelles ne peuvent pas être allouées au même serveur.
- B- Domaine d'ordonnancement :** (localité d'ordonnancement) Un domaine est constitué d'un ou plusieurs serveurs physiques ou clusters. Certains algorithmes de planification, comme l'équilibrage de charge ou l'utilisation maximale, ne peuvent se produire qu'en un seul domaine de planification au sein du système de planification. L'ordonnancement dans plusieurs domaines est possible et est très important, alors que parfois une opération manuelle est nécessaire. Du point de vue des couches, un domaine de planification ne peut se produire qu'au niveau de la couche système ou plus petit (bien sûr, il peut s'agir d'un seul domaine de planification dans un centre de données).
- C- Temps :** (temps disponible limité) Les tâches allouées à la même ressource ne peuvent pas partager le même temps d'exécution. Si une machine virtuelle occupe une ressource pendant un certain temps, les autres utilisateurs ne peuvent pas utiliser la même ressource en même temps, sauf si elles ont une priorité plus élevée dans le domaine de l'ordonnancement.

II.8. Ordonnancement de temps d'exécution et les conditions du déclenchement des tâches

Lorsque les coûts des ressources sont pris en compte, tous les scénarios ne déclenchent pas la planificatrice pour répondre instantanément. Par exemple, si le moniteur remarque que l'utilisation du processeur a dépassé le seuil, cela pourrait être juste temporaire, donc un calendrier instantané ne serait pas raisonnable. La planification d'optimisation commune adopte généralement un calendrier cyclique. Pour les situations inhabituelles, comme lorsqu'une machine est en panne - le planificateur se déclencherait instantanément. [20]

Une condition de planification peut être classée dans l'un des trois types suivants :

- 1. Performances de planification régulières :* tient compte de la demande de l'entreprise lors de la planification de l'exécution le temps est défini (c'est-à-dire à intervalles réguliers ; si la demande commerciale est élevée, il peut être défini sur 1 min. Si la demande des entreprises est faible, elle peut être réglée sur 10 min).
- 2. Condition peu commune :* se produit lorsque les ressources physiques sont rompues ou que l'équilibre de charge a dépassé le seuil prédéfini.
- 3. Optimisation avec fonctionnement manuel :* organise le programme manuellement.

II.9. Exemple d'une stratégie d'ordonnancement

Il est très important de choisir l'efficace stratégie d'ordonnancement pour s'adapter aux différentes activités conditions. Les propriétaires et gestionnaires de cloud doivent définir les stratégies de planification pour gérer leur ressource et satisfaire les besoins des utilisateurs. Chaque entreprise sélectionne l'algorithme d'ordonnancement qui satisfait sa stratégie conditions. [1]

II.8.1. Amazone

La stratégie de planification du cloud computing d'Amazon combine des stratégies axées sur le coût d'abord, répond alors aux différentes exigences des utilisateurs comme l'équilibrage de charge ou la haute fiabilité :

1. Coûts différentiels : Il utilise une stratégie d'allocation de planification des charges qui implique différentes régions ayant des coûts différents parmi lesquels les clients peuvent choisir.

2. Accélération de la vitesse de réponse : préconfigure les applications de machine virtuelle typiques.

3. Classification des entreprises : les utilisateurs peuvent être classés immédiatement par type d'entreprise utilisant le client, réservations et autres informations. Les frais diffèrent et les frais du client de réservation sont un peu inférieurs aux frais d'utilisation immédiate des clients.

4. Norme de tarification générale : les frais d'utilisation à long terme par unité de temps sont inférieurs à ceux à court terme frais d'utilisation.

5. Équilibrage de charge : Round Robin est l'une des technologies de stratégie de planification. C'est seulement nécessité une fonction circulaire de liste d'adresses IP, par opposition aux adresses IP uniques traditionnelles fourni pour l'utilisation du DNS. Le serveur répond aux requêtes, fournit la première adresse IP à la première requête, la deuxième adresse IP à la requête suivante, et répète le processus jusqu'à l'IP finale est distribuée. Ensuite, le serveur répète ce processus. Round Robin est très adapté aux serveurs dans différentes zones ou aux serveurs avec des contenus dans plusieurs centres de données ou les serveurs.

II.10. Problèmes de l'ordonnancement dans le cloud

Nous déjà déclaré dans le titre précédent (II.6), que Le problème d'ordonnancement peut être classé en deux catégories en fonction de l'objet O : problème d'optimisation et problème de décision. Un problème d'optimisation nécessite de trouver la meilleure solution parmi toutes les

solutions réalisables dans l'ensemble S . Différent de l'optimisation, le but du problème de décision est relativement facile. Pour une solution réalisable spécifiée $s \in S$, le problème a besoin d'une réponse positive ou négative pour savoir si l'objet O est atteint. De toute évidence, le problème d'optimisation est plus difficile que le problème de décision, car la solution spécifiée ne se compare qu'à une solution de seuil dans le problème de décision, au lieu de toutes les solutions réalisables dans le problème d'optimisation.

II.11. Optimisation et résoudre les problèmes de l'ordonnancement dans les clouds avec logique flou

II.11.1. Etat de l'art et travaux connexes

Le tableau suivant résume quelque travaux connexe et lies directement avec notre étude, les six (06) articles présentées traité du manière académique les challenges et les problèmes lies avec la stratégie de l'ordonnancement, donc l'ordonnancement des taches et propose des solutions d'optimisation selon les algorithmes hybride, cette dernière technique utilisant logique flou (Fuzzy logic) pour obtenir un ordonnancement optimal.

Tableau II. 3: Les études académiques connexes

Article et Année	Contributions	Avantages	Logiciels	Travaux Future
[01] 2013	Présenté une nouvelle approche pour l'algorithme d'ordonnancement dynamique sur le Cloud, appelée l'ordonnancement flou, utilisant le contrôleur de logique floue.	Présentation de l'efficacité de l'algorithme hybride en le comparant à d'autres algorithmes plus anciens, aussi ; réduire le temps d'achèvement	CloudSim simulator pour générer des Cloudlets; Excel pour les tableaux et les graphes	/
[02] 2014	Modifier l'algorithme génétique standard et réduire l'itération de création de population à l'aide de la théorie floue.	Affectez les travaux aux ressources en tenant compte du MIPS de la machine virtuelle et des Tailles des tâches. Améliore les performances du système en termes de coût d'exécution et de temps d'exécution total.	Origin Pro (pour les tableaux et les graphes) ; Matlab pour les Fuzzy graphes.	/
[03] 2015	Présente et examine un nouvel algorithme Hybride de planification avec l'utilisation d'un système de contrôle flou pour la planification de machines virtuelles entre les centres de données (DATAcenters).	Réduire le temps de réponse des tâches Il montre également de meilleures performances que le Algorithme anciennes		/
[04] 2020	Algorithme méta-heuristique hybride utilisant la logique floue pour calculer l'aptitude des organismes en combinant l'algorithme de recherche d'organismes symbiotiques avec la logique floue, appelée FLSOS	La minimisation du MakeSpan et l'amélioration du degré d'équilibre entre les machines virtuelles	CloudSim	/
[05] 2020	- Détermination du coefficient de contrainte de fiabilité ρ dans l'algorithme FR-MOS proposé utilisant la logique floue avec l'utilisation des ressources.	FR-MOS surpasse le MOS de base par rapport à l'algorithme PSO.	Trois Clouds commerciaux (Amazon EC2, Google Compute Engine et Microsoft Azure) et leurs VM. - Workflowsim 1.0.	Etendre la stratégie de planification des flux de travail pour atteindre la tolérance aux pannes dans un environnement multi-cloud. Une concession également aux perspectives de planification des flux

	<ul style="list-style-type: none"> - Intégration de la méthode PSO avec l'algorithme FR-MOS. - Application de l'algorithme FR-MOS proposé sur quatre workflows scientifiques réels. 		<ul style="list-style-type: none"> - PC i7 (6 cores) avec 16 G de RAM. - scientific workflows (Montage, LIGO, SIPHT et CyberShake). 	de travail pour réduire la consommation d'énergie dans un environnement de cloud hybride.
[06] 2022	Améliorer la planification des tâches dans le cloud computing, selon des règles floues.	Différents algorithmes méta-heuristiques tels que l'algorithme génétique, PSO, ACO, ABC et PACO sont développés pour la planification des tâches dans l'environnement de cloud computing	CloudSim	Optimisant un nombre plus important d'objectifs tels que la disponibilité des tâches, l'efficacité énergétique, la qualité de service complète de l'utilisateur, etc. , qui peut fournir les meilleurs résultats optimisés. (Cat Swarm Optimization, Cockroach Swarm Optimization, Bean Optimization)

II.10.2. Notions sur la logiques floue

A) Logique floue

La logique floue traite du raisonnement avec des concepts inexacts ou flous. Par conséquent, les isomorphismes bien établis entre l'algèbre booléenne³, la théorie des ensembles et la logique propositionnelle peuvent être étendus de manière naturelle entre l'algèbre floue, la théorie des ensembles flous et la logique floue. [23]

II.11. Application de logique floue dans les problèmes de l'ordonnancement

La logique floue n'a pas d'affectation stricte d'éléments à des ensembles comme la logique binaire. Au lieu de cela, chaque élément a un degré d'appartenance à un ensemble. Ce degré est représenté par une valeur comprise entre 0 et 1. Pour pouvoir appliquer la logique floue à un problème spécifique tel que l'ordonnancement entre cloudlets ou machines virtuelles, un système flou doit être construit [2] :

A- Fuzzification : Dans cette étape, le degré d'appartenance des valeurs d'entrée est affecté à des ensembles flous. Le degré d'appartenance est donné par : $\mu_X \rightarrow [0 \quad 1]$; où *X est*

³ - Les mathématiques floues sont une branche des mathématiques qui contient la théorie des ensembles flous et la logique floue. En 1965, Lotfi A. Zadeh a introduit les concepts Fuzzy. (Voir : Dictionnaire mathématique Online)

l'ensemble des valeurs d'entrée. Ainsi, chaque valeur d'entrée est mappée à une valeur comprise entre zéro et un.

B- Défuzzification : dans cette étape, une valeur de sortie numérique est générée à partir de l'ensemble de sorties.

C- Les systèmes d'inférence floue (FIS) : sont conceptuellement simples. Ils se composent d'un étage d'entrée, d'un étage de traitement et d'un étage de sortie. L'étape d'entrée mappe les entrées, telles que la fréquence de référence, la récence de la référence, etc., aux fonctions d'appartenance et aux valeurs de vérité appropriées. L'étape de traitement invoque chaque règle appropriée et génère un résultat correspondant. Il combine ensuite les résultats. Enfin, l'étape de sortie reconvertit le résultat combiné en une valeur de sortie spécifique. [24]

Différentes interprétations des règles floues if-then⁴ (SI-ALORS), y compris les implications de *Dienes-Rescher, Lukasiewicz, Zadeh, Godel et Mamdani*. [25]

⁴ - If q then p, où p et q sont des variables propositionnelles dont les valeurs sont soit vrai (T) soit faux (F). (Voire : [25])

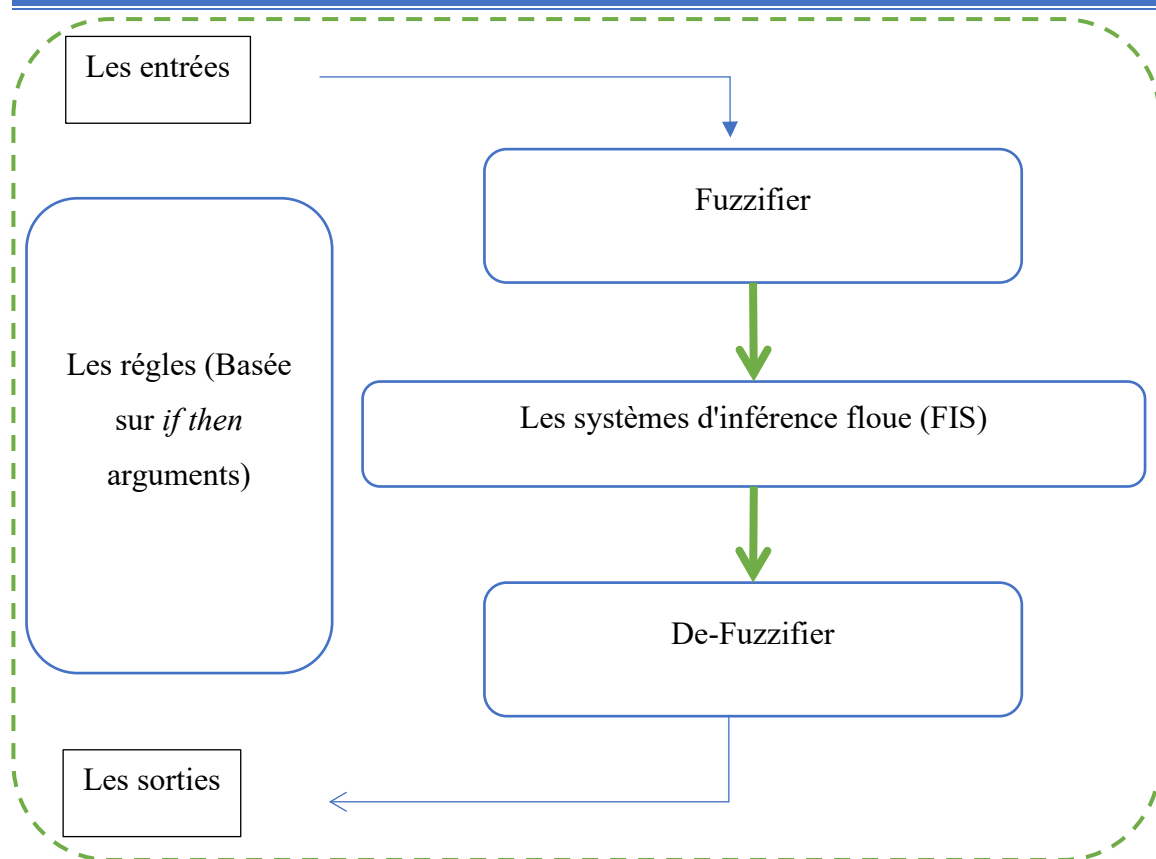


Fig II. 3: Structure du contrôleur de logique floue [25]

Selon la fig II.3, Le concept de commande floue consiste à décrire des algorithmes basés sur des règles logiques. Les règles sont soit obtenues à partir des connaissances d'experts, soit calculées à partir de mesures numériques. La méthodologie de la commande floue repose sur la définition et l'utilisation [26]:

- un ensemble de règles illustrant la stratégie de contrôle ;
- fonctions d'appartenance par règle ;
- connecteurs logiques pour les relations floues ;
- méthode de défuzzification.

II.12. Conclusion

Nous avons sous-adjacent quelque titres principaux lies à l'ordonnancement et Logique floue (fuzzy logic).

D'autre part, l'ordonnancement et les problèmes de planifications est une Domain très large et intense ; a conséquence plusieurs chercheurs académique travaillant sur ce sujet ; donc comme déjà cité, nous avons traitées les titres avec caution, et les titres qui sont directement liés à l'ordonnancements et logique floue dans les clouds du coté de Planification des taches (Ressources et stratège) avec les algorithmes hybridés.

Chapitre III

Conception d'un ordonnanceur basé sur la logique floue dans le Cloud

III.1. Introduction

Dans les deux premiers chapitres, nous présentons l'essentiel des notions de base sur le cloud computing et l'ordonnancement dans l'environnement cloud .

En s'appuyant sur quelques travaux de la littérature qui traitent le problème d'ordonnancement dans les clouds avec logique flou, nous proposons un système d'ordonnancement des tâches en utilisant la logique floue. Les entrées et les sorties de système flou doivent être les premiers choisis. Après cela, ils doivent être partitionnés en catégories conceptuelles qui représentent en fait un ensemble flou sur un domaine d'entrée ou de sortie donné. Les conceptuelles développés pour les entrées et les sorties sont utilisés pour créer un ensemble de règles floues qui détermine le comportement du flou système

Ce chapitre est consacré à l'explication de notre principale contribution et à la modélisation de notre système. Suite à la présentation de notre contribution, nous détaillerons les éléments de conception de l'approche proposée en définissant chaque modèle.

III.2. Notre contribution

Nous proposons un modèle d'optimisation base sur logique floue adressé le niveau d'ordonnancement de tâche dans le cloud computing ; On a utilisé un algorithme d'ordonnancement avec logique floue, pour tester et évaluée ces algorithmes de la première chose et deuxièmement prouvée l'importance de logique floue dans ce cas des problèmes.

III.2.1. Problématique

Le cloud computing continue de convaincre chaque jour de nouveaux utilisateurs issus des secteurs scientifiques et industriels. Pour satisfaire les différents besoins des utilisateurs, les fournisseurs de Cloud doivent maximiser les performances de leurs ressources informatiques pour assurer le meilleur service au moindre coût. Les efforts d'optimisation des performances dans le Cloud peuvent être réalisés à différents niveaux et aspects. Mais Le challenge real définir par le problème de l'optimisation dans plusieurs domaines tels que l'exploration de données, le traitement d'images, la mise en réseau et bien plus encore l'ordonnancement des tâches. En effet, comment optimiser la planification des tâches dans les clouds sans modifier les propriétés physiques du cluster, c'est-à-dire ; réduire le temps passé par les ressources du cloud pour effectuer plusieurs tâches?

III.2.2. Objectifs

Nous proposons d'introduire un processus de logique floue dans la stratégie d'ordonnancement du Cloud public afin d'optimiser l'utilisation des ressources, ce qui réduit le temps de réponse moyen du système.

III.2.3. Modèle proposé

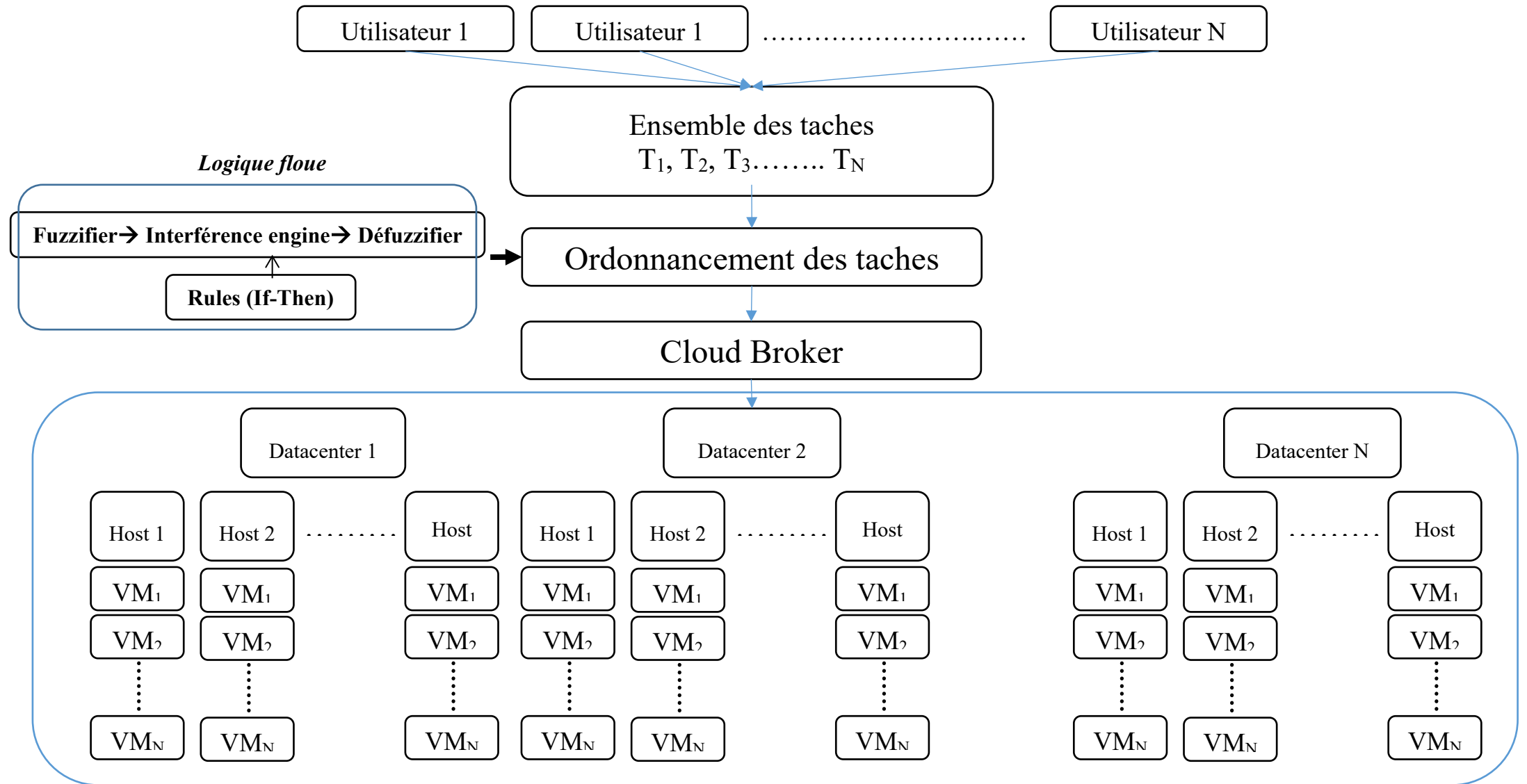


Fig III. 1: Schéma générale de système proposé

Datacenter: les services de base au niveau de l'infrastructure (c'est-à-dire le matériel) proposés par les fournisseurs de cloud (Amazon, Azure et App Engine). Il encapsule un ensemble d'instances d'hôtes (ressemblant à un modèle de machine serveur) qui peuvent être homogènes ou hétérogènes en ce qui concerne leurs configurations matérielles (mémoire, cœurs, capacité et stockage). En outre, chaque composant Datacenter prend en charge le provisionnement généralisé des applications qui applique un ensemble de politiques pour l'allocation de la bande passante, de la mémoire et des périphériques de stockage aux hôtes et aux machines virtuelles associées.

Datacenter Broker or Cloud Broker: est responsable de la médiation des négociations entre les fournisseurs SaaS et Cloud et ces négociations sont motivées par les exigences de QoS. La classe Broker agit au nom des applications. Son rôle principal est d'interroger le CIS pour découvrir les ressources/services appropriés et entreprendre des négociations pour l'allocation des ressources/services qui peuvent répondre aux besoins de QoS de l'application. Cette classe doit être étendue pour évaluer et tester les stratégies de courtage personnalisées.

Host: est une ressource physique telle qu'un ordinateur ou un serveur de stockage. Il encapsule des informations importantes telles que la quantité de mémoire et de stockage, une liste et le type de cœurs de traitement (s'il s'agit d'une machine multicœur), une allocation de politique pour le provisionnement du calcul, de la mémoire et de la bande passante aux machines virtuelles.

Vm: est gérée et hébergée par un composant hôte Cloud. Chaque composant de machine virtuelle a accès à un composant qui stocke les caractéristiques suivantes liées à une machine virtuelle (c'est-à-dire la mémoire accessible, le processeur, la taille de stockage et la politique de provisionnement interne de la machine virtuelle).

III.2.4. Caractéristiques de modèle proposé

Le modèle proposé consiste à l'injection de fuzzy système dans un algorithme d'ordonnancement. L'implémentation Logique floue sera faite avec les considérations suivantes

- Des taches no-périodique.
- Une seul CPU partagée sur plusieurs utilisateurs.
- FCFS est par défaut l'algorithme d'ordonnancement.

III.3. La méthode d'ordonnancement proposée avec la logique floue

Les étapes de la logique floue sont expliquées dans cette section : Tout d'abord, un ensemble précis de données d'entrée est collecté et converti en un ensemble flou à l'aide de variables linguistiques, de termes linguistiques et de fonctions d'appartenance. Cette étape est connue sous le nom de fuzzification. Ensuite, une inférence est préparée sur la base d'un ensemble de règles. Enfin, la sortie floue est mappée sur une sortie nette à l'aide des fonctions d'appartenance, lors de l'étape de défuzzification.

. L'algorithme proposé considère nombre de PE, Host Storage, and RAM comme paramètres d'information pour mesurer la priorité d'un VM (Virtual machine).



Fig III. 2: les entrés et les sorties de système de logique floue

Les variables d'entrée sont mappées par un ensemble de fonctions d'appartenance dans le système de logique floue. Le fait de changer la valeur d'entrée en valeur floue est appelé fuzzification. La fuzzification dans le système de logique floue serait basée sur les règles et la défuzzification. Après la défuzzification, nous obtiendrons une sortie unique pour le nombre d'entrées spécifié. Dans la figure 2, nous fournissons nombre de PE ,Host storage, and RAM comme entrée du système de logique floue. La sortie du système de logique floue est la priorité de machine virtuelle basé sur le système de logique floue. Les règles utilisées dans le système flou sont représentées dans le Tableau. III 1.

Tableau III. 1: les règles utilisées dans le système de logique floue

Les règles	Les paramètres d'entrée			Les paramètres de sortie
	RAM	host Storage	PE	priorité
1	Low	Low	Low	Low
2	Low	Low	medium	Low
3	Low	Low	High	Low
4	Low	medium	Low	Low
5	Low	medium	medium	Low
6	Low	medium	High	Low
7	Low	High	Low	Low
8	Low	High	medium	Low
9	Low	High	High	Low
10	medium	Low	Low	Low
11	medium	Low	medium	Low
12	medium	Low	High	Low
13	medium	medium	Low	Low
14	medium	medium	medium	medium
15	medium	medium	High	medium
16	medium	High	Low	Low
17	medium	High	medium	medium
18	medium	High	High	medium
19	High	Low	Low	Low
20	High	Low	medium	Low
21	High	Low	High	Low
22	High	medium	Low	Low
23	High	medium	medium	medium
24	High	medium	High	medium
25	High	High	Low	Low
26	High	High	medium	medium
27	High	High	High	High

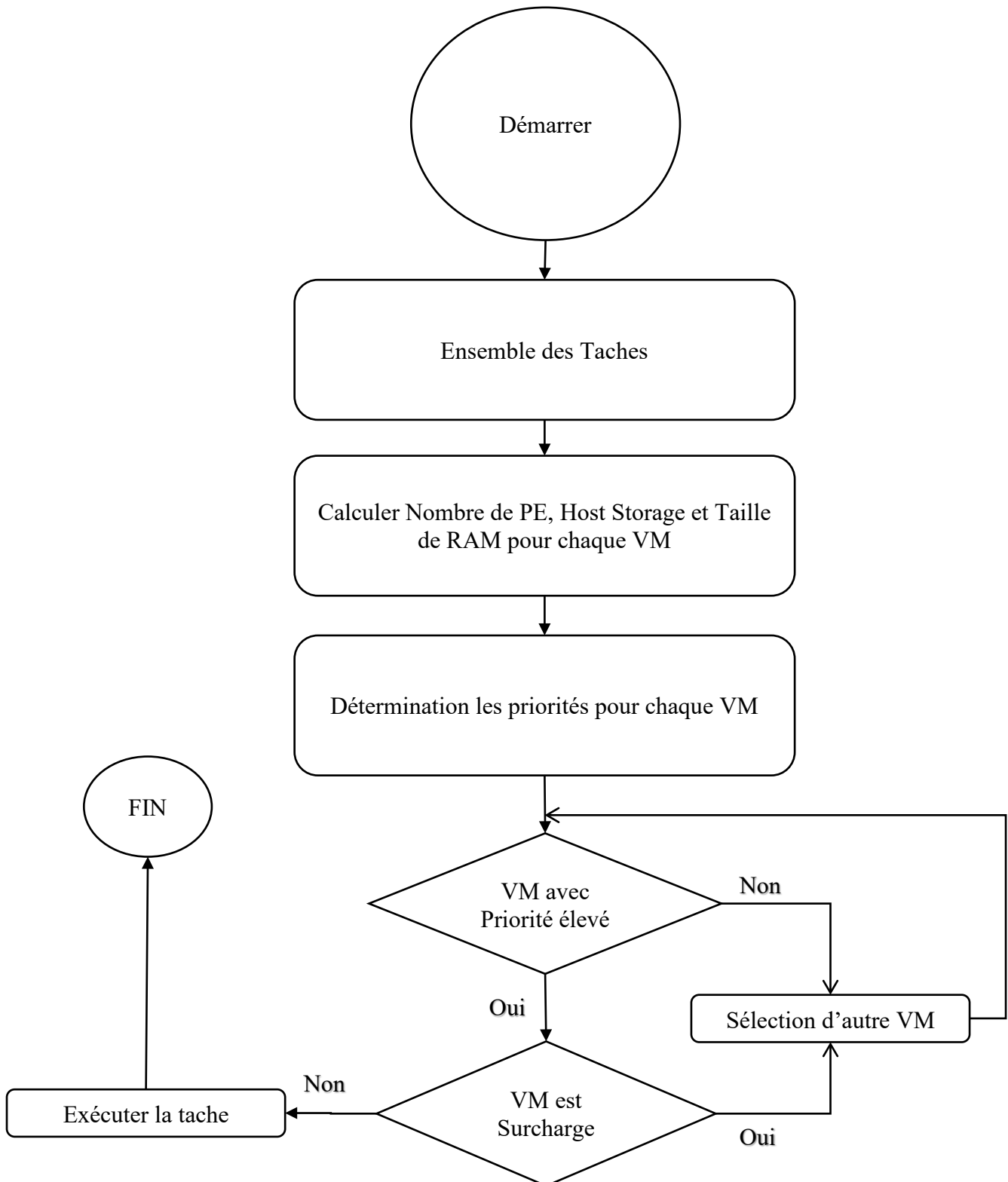


Fig III. 3: Diagramme d'activité de système de logique floue proposé

III.4. Conclusion

Dans ce chapitre, nous avons proposé une nouvelle approche d'optimisation basée sur l'hybridation avec la logique floue pour l'ordonnancement des tâches dans l'environnement cloud. Nous proposons une solution sous la forme d'une simulation d'un cloud à l'aide de CloudSim.

Nous avons remarqué d'après les résultats obtenus par le simulateur CloudSim que notre algorithme a réussi à améliorer du manière acceptable les trois objectifs par rapport Les autre algorithmes.

Chapitre IV:

**Implémentation d'un algorithme
d'ordonnancement hybridée avec logique floue**

IV.1. Introduction

Le simulateur CloudSim, construit par le Laboratoire CLOUDS à Melbourne, Australie, est l'instrument d'évaluation sélectionné pour cette étude.

Les clouds ou Grid Computing (Aspect généralisé) est confrontée à un problème important dans l'utilisation efficace des ressources telles que la mémoire, les réseaux de communication pendant l'ordonnements ou la planification des taches demandées par des utilisateurs, les cycles de calcul et les référentiels de données afin d'atteindre les objectifs souhaités. La découverte des ressources disponibles et la sélection des ressources acceptables sont essentielles en raison de l'hétérogénéité des ressources et de la variation des applications.

Une base pour représenter des connaissances imprécises et incertaines est fournie par la logique floue. Les systèmes flous utilisent une sorte de raisonnement par approximation, qui leur permet de traiter des informations vagues et incomplètes, similaires à la façon dont les humains prennent des décisions.

Dans ce chapitre notre buts sont :

- ✓ Présentation et utilisation CloudSim et l'IDE propre avec les Package.
- ✓ Exécution de programme.

IV.2. Outils de développement

IV.2.1 Environnement machine

Notre étude dépend fortement de l'amélioration du logiciel, mais nous ne pouvons pas ignorer les propriétés physiques de la machine que nous avons utilisée.

Nous avons utilisé un PC de bureau à processeur AMD, type Ryzen 3 (3400G) avec 24 G de RAM.

IV.2.2. Environnement logiciel

Tout projet informatique nécessite l'utilisation de technologies performantes afin d'assurer une bonne implémentation des besoins définis dans les phases précédentes. Et pour le développement de cette application, nous avons utilisé les outils logiciels suivants :

CloudSim :

CloudSim⁵, projet développé par le CLOUDS Laboratory de Melbourne en Australie, est un outil de simulation extensible permettant la modélisation et la simulation d'environnement de systèmes de type Cloud Computing de niveau IaaS. Cette section d'introduction aux fonctionnalités de CloudSim est largement inspirée de l'article présentant en détails ce simulateur [26].

Librairie Fuzzy utilisée :

Module de : *Cloudbus Cloudsim Fuzzylogic* modifiée, elle contient les class suivantes :

Defuzzifier : classe pour arranger les outputs.

Fuzzifier : classe destinée a arrangé les inputs.

FuzzyInferenceSystem : Classe pour générer des priorités pour les cloudlets (initFISRules : Fuzzifier, Defuzzifier)

IV.3. Architecture générale

Illustrée en Figure 3.1, son architecture permet la modélisation des différents composants d'un Cloud: Data Center, machines physiques, machines virtuelles etc... La gestion de toutes ces ressources est gérée par la couche Cloud Services, gérant l'approvisionnement des machines virtuelles, l'utilisation des CPUs, de la mémoire RAM, de la capacité de stockage et de la bande

⁵ - Il exist également CloudSim Plus, est un framework de simulation Java 17 moderne, à jour, complet et entièrement documenté. Il permet aux développeurs de se concentrer sur des problèmes de conception de système spécifiques à étudier, sans se soucier des détails de bas niveau liés aux infrastructures et services basés sur le cloud. (Voir <https://cloudsimplus.org/>)

passante des machines physiques. Dans CloudSim, les tâches à exécuter sont assimilées à des cloudlets qui sont alloués aux machines virtuelles (couches VM Services et User Interface Structure). La configuration de chacun des composants doit être gérée par l'utilisateur (couche Simulation Spécification) par l'intermédiaire d'un fichier de configuration de simulation. De nombreux paramètres peuvent y être spécifiés :

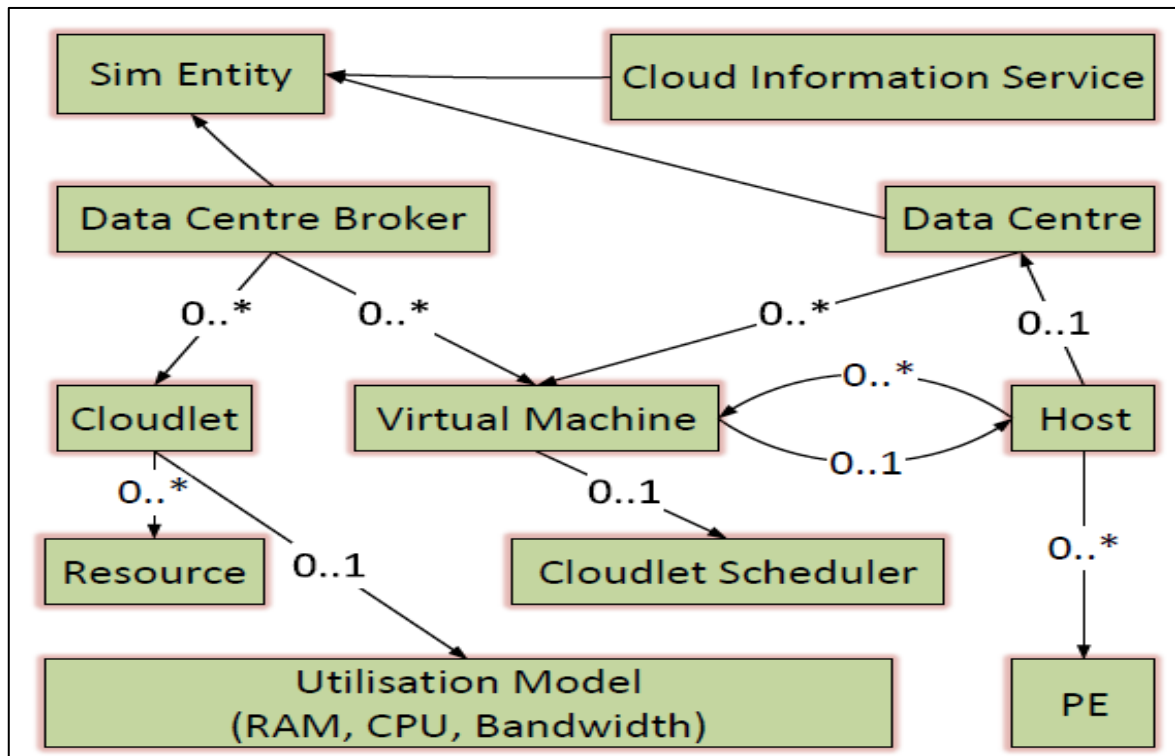


Fig. IV 1: Architecture CloudSim pour l'ordonnancement [27].

- Nombre de Data Center, machines physiques, machines virtuelles, tâches (cloudlets)
- **Caractéristiques des Data Center** : architecture, système d'exploitation, hyperviseur, coûts de fonctionnement
- **Caractéristiques des machines physiques** : nombre de CPUs, capacité des CPUs, capacité mémoire, Capacité de stockage, bande passante
- **Caractéristiques des machines virtuelles** : capacité CPUs, nombre de CPUs, capacité mémoire, capacité de stockage
- **Caractéristiques des cloudlets** : nombre d'instructions à exécuter, taille des fichiers d'entrée et de Sortie.

IV.4. Métriques de performances

La mesure de performance utilisée pour l'évaluation de l'efficacité de notre algorithme d'ordonnancement est le makespan.

Le Makespan est le temps de terminaison de l'exécution de toute la simulation en d'autres termes, c'est le temps de la fin d'exécution de la dernière cloudlet de tous les cloudets.

IV.5. Résultats de simulation

Le simulateur utilisé dans notre travail est une extension version de CloudSim appelé "FuzzyCloudsim". Afin de valider et d'évaluer nos algorithmes, nous avons effectué une série des simulations. On va présenter et comparer les résultats de différentes simulations avec une discussion à la fin de cette section.

Simulation 1 :

Dans la première simulation, nous avons choisi l'algorithme FCFS comme un algorithme d'ordonnancement par défaut, nous prenons différents nombre de cloudets et VM et nous avons mesuré Makspan comme le montre le tableau IV.1.

```
===== OUTPUT =====
```

Cloudlet ID	STATUS	Data center ID	VM ID	Time	Start Time	Finish Time
0	SUCCESS	2	0	800	0,4	800,4
4	SUCCESS	2	0	800	0,4	800,4
8	SUCCESS	2	0	800	0,4	800,4
12	SUCCESS	2	0	800	0,4	800,4
16	SUCCESS	2	0	800	0,4	800,4
1	SUCCESS	3	1	800	0,4	800,4
5	SUCCESS	3	1	800	0,4	800,4
9	SUCCESS	3	1	800	0,4	800,4
13	SUCCESS	3	1	800	0,4	800,4
17	SUCCESS	3	1	800	0,4	800,4
2	SUCCESS	4	2	800	0,4	800,4
6	SUCCESS	4	2	800	0,4	800,4
10	SUCCESS	4	2	800	0,4	800,4
14	SUCCESS	4	2	800	0,4	800,4
18	SUCCESS	4	2	800	0,4	800,4
3	SUCCESS	5	3	800	0,4	800,4
7	SUCCESS	5	3	800	0,4	800,4
11	SUCCESS	5	3	800	0,4	800,4
15	SUCCESS	5	3	800	0,4	800,4
19	SUCCESS	5	3	800	0,4	800,4

Fig. IV 2: Exemple d'exécution de simulation 1

Tableau IV 1: le Makspan (en nanosecondes) avec différents nombres of cloudets et VM en utilisant FCFS

Nombre de datacenter	Nombre de VM	Nombre de cloudets	Makspan
4	2	100	50.1
4	3	500	165.84
4	4	1000	250
4	6	2000	333.67
4	10	3000	299.8
4	20	4000	200.1
2	2	100	50.1
2	3	500	160.84
2	4	1000	333.67
2	6	2000	399.8
2	10	3000	333.77
2	20	4000	266.74

Simulation 2 :

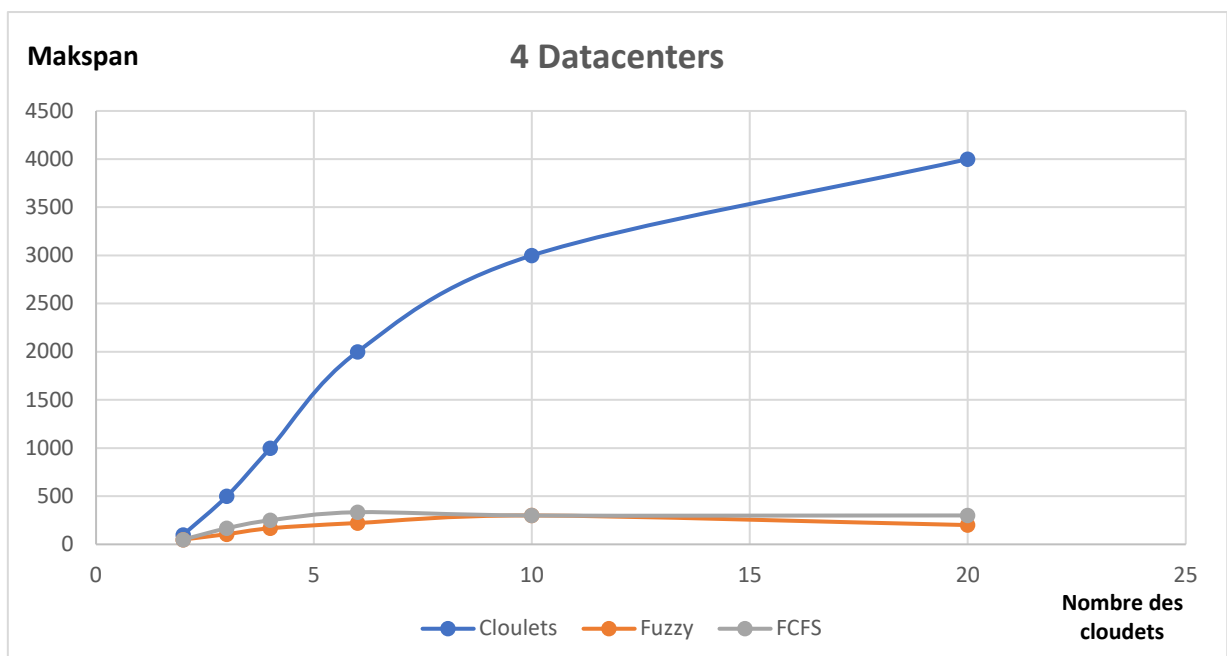
Dans la deuxième simulation, nous avons utilisé l'algorithme proposé (fuzzy logic), nous prenons différents nombres de cloudets et VM et nous avons mesuré Makspan comme le montre le tableau IV.2.

Tableau IV 2: le Makspan (en nanosecondes) avec différents nombres of cloudets et VM en utilisant l'algorithme proposé

Nombre de datacenter	Nombre de VM	Nombre de cloudets	Makspan
4	2	100	50.1
4	3	500	166.84
4	4	1000	250
4	6	2000	333.67
4	10	3000	300
4	20	4000	200.4
2	2	100	50.1
2	3	500	160.48
2	4	1000	366.00
2	6	2000	401.81
2	10	3000	366.00
2	20	4000	333.87

Simulation 3 :

Dans la troisième simulation, nous avons comparé l'algorithme proposé (fuzzy logic), avec l'algorithme FCFS, nous prenons différents nombres de cloudets et VM et nous avons comparé Makspan comme le montre la fig. IV.3.



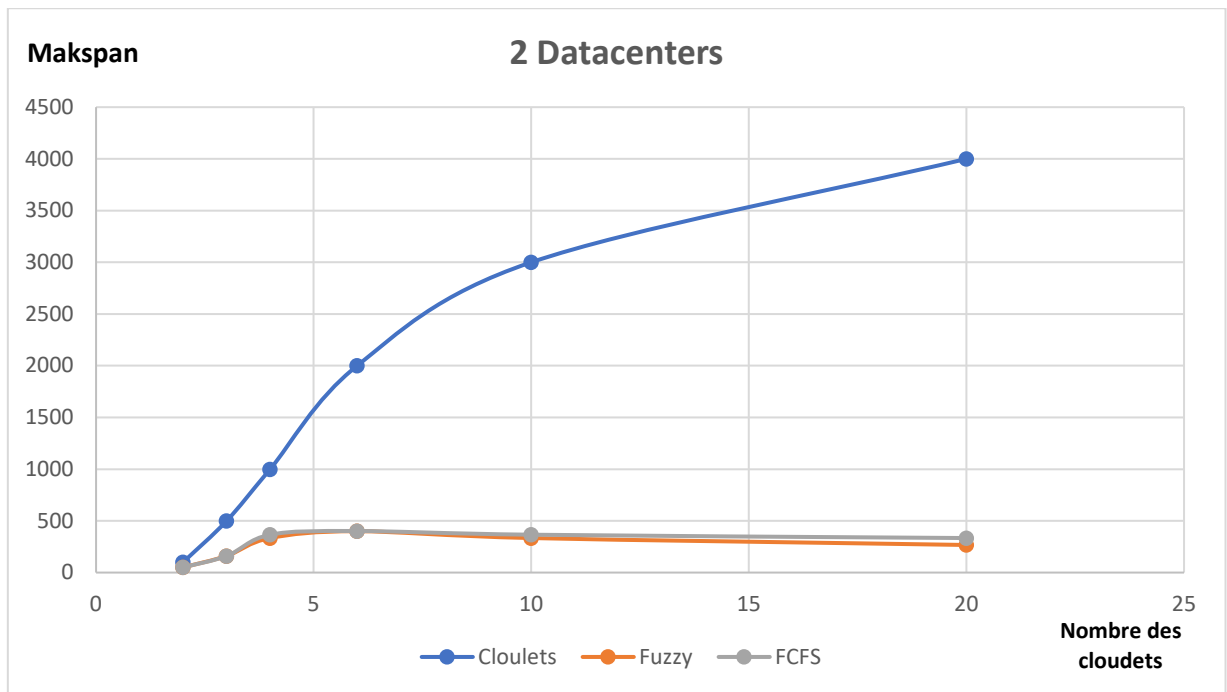


Fig. IV 3: Comparaison le Makspan entre l'algorithme proposé et FCFS

IV.6. Conclusion

Nous allons passer en revue le langage de programmation Java, l'environnement de développement IntelliJIdea, et pourquoi nous avons choisi ces deux outils dans ce chapitre. Après cela, le simulateur CloudSim est discuté, y compris son architecture et ses classes de base, ainsi que les classes que nous avons incorporées. Ensuite, nous passerons en revue les nombreux processus et configurations qui doivent être complétés avant de pouvoir commencer à simuler nos techniques.

Après la première simulation, nous avons indiqué les résultats de chaque makespan qui correspondent à des valeurs différentes pour VM, pour Cloudlet et différentes valeurs pour les Datacenters selon l'algorithme FCFS. la deuxième simulation, nous avons répété les mêmes procédures mais nous avons changé l'algorithme pour un nouveau, un FCFS basé sur la logique floue. une comparaison entre les deux algorithmes indique qu'il existe une différence notable entre eux, l'algorithme de logique floue prend moins de temps pour ordonnancer les cloudlets au sein des Datacenters vers les VM.

L'algorithme FCFS ne fait pas la distinction entre les tâches, il y a donc toujours une chance qu'une tâche avec une faible priorité et un temps long soit exécuté en premier et prenne beaucoup de temps, cela ralentira tout le système de planification, en revanche, et avec la présence de logique floue ; la priorité de toutes les tâches entrantes sera réorganisée en fonction de sa priorité, à la fin, le système de l'ordonnancement sera optimisé et fonctionnera mieux.

Conclusion Générale

Conclusion Générale

L'un des aspects les plus difficiles du cloud computing est de déterminer le meilleur moment pour planifier les tâches. L'ordonnancement est l'acte de définir l'ordre dans lequel les tâches des différents processus doivent être accomplies et les ressources que chacun d'eux doit employer. De plus, les utilisateurs sont confrontés au défi de décider quelles ressources (fournisseurs) utiliser.

Dans le domaine du cloud computing, le problème d'ordonnancement reste entier. De plus, il n'existe aucune technique d'ordonnancement qui résout le problème d'une tâche s'exécutant simultanément avec les ressources disponibles. [14]

La difficulté de planification des tâches dans le cloud computing a été examinée dans notre recherche. Ce problème est classé comme un problème d'optimisation multi-objectif dans la littérature. Dans cette mémoire, nous avons traité la problématique de comment optimiser les performances du cloud dans l'aspect et le niveau de planification des tâches d'une manière; que les utilisateurs puissent bénéficier des services des nuages en peu de temps.

Pour achever ces objectifs, nous avons présenté et évalué un nouvel algorithme de planification basé sur la logique floue en tant que technique efficace de planification des machines virtuelles entre les Datacenter afin d'obtenir la meilleure priorité des tâches demandées par les utilisateurs.

Après l'analyse des résultats obtenus, nous avons noté que les résultats de la simulation montrent l'efficacité de notre algorithme en le comparant aux techniques d'ordonnancement (FCFS). Ainsi, l'implémentation de logique floue a amélioré l'algorithme classique en créant un hybride plus efficace. Les résultats montrent que l'algorithme d'ordonnancement proposé peut être efficacement déployé sur les Clouds.

Perspectives

Dans notre étude, nous avons présenté une petite partie de l'algorithme basé sur logique floue destinées à l'ordonnancement des tâches dans les Clouds, mais c'est un domaine énorme qui nécessite encore des années de recherche et d'étude, Diverses questions ouvertes sont également abordées dans notre document qui peuvent être reprises pour de futures recherches. Donc ; peut-être qu'à l'avenir, les études concentreront sur les algorithmes hybrides avec 3 algorithmes différents ou plus comme (Particle Swarm Optimization [PSO], Round Robin [RR], Random Forest [RF]).

Par ailleurs, les études doivent être étendues aux techniques d'ordonnancement dynamique, comme les algorithmes méta-heuristiques pures ou hybrides.

Bibliographie

Bibliographie

- [1] S. A. E. Aida Abou-Elseoud Nasr, "Scheduling Strategies in Cloud Computing: Methods and Implementations," *American Journal of Engineering and Applied Sciences*, pp. 1-8, 2018.
- [2] S. M. H. Amin Mehranzadeh, "A Novel-Scheduling Algorithm for Cloud Computing based on Fuzzy Logic," *International Journal of Applied Information Systems (IJ AIS)*, vol. 05, no. 07, pp. 28-31, 2013.
- [3] M. Sunilkumar and S. K. Gopal, *Cloud Computing concepts and technologies*, USA: CRC Press, 2021, pp. 1-2.
- [4] A. Mehranzadeh, "A Novel-Scheduling Algorithm for Cloud Computing based on Fuzzy Logic," *IEEE*, vol. 5, no. 7, pp. 28-31, 2013.
- [5] A. Lisdorf, *CLOUD COMPUTING*, Denmark: APRESS, 2021.
- [6] S. M. S. D. A. N. C. H. L. a. A. A. Javanmardi, "Hybrid Job Scheduling Algorithm for Cloud Computing Environment," vol. 3, no. 3, pp. 43-52, 2014.
- [7] K. Anindita, "A New Approach for Task Scheduling of Cloud Computing Using Fuzzy," *International Journal of Innovative Research in Computer Science & Technology (IJIRCST)*, pp. 112-116, 2015.
- [8] T. G. Peter Mell, "The NIST Definition of Cloud Computing," *NIST Special Publication 800-145*, pp. 1-7, septembre 2011.
- [9] B. B. Zouheyr Bouafia, "Algorithmes d'ordonnancement des tâches dans un environnement Cloud," *Revue Méditerranéenne des Télécommunications*, vol. 05, no. 02, pp. 71-78, 2015.
- [10] D. E. COMER, *The Cloud Computing*, USA: CRC press, 2021.
- [11] "RedHat Linux," Linux inc, [Online]. Available: <https://www.redhat.com/fr/topics/cloud-computing>. [Accessed 19 02 2022].
- [12] I. I. center, *Infrastructure des clouds*, USA: Intel, 2014.
- [13] Z. M. R. P. Thomas Erl, *Cloud Computing Concepts, Technology & Architecture*, Massachusetts.: Arcitura Education Inc, 2013.
- [14] R. N. Anis VOSOOGH, "Scheduling Problems For Cloud Computing," *Science Journal*, vol. 36, no. 3, pp. 2629-2652, 2015.
- [15] J. B. A. G. Rajkumar Buyya, *CLOUD COMPUTING Principles and Paradigms*, USA: Wiley, 2011.
- [16] D. E. Insaf, *OPTIMISATION D'ORDONNANCEMENT ET D'ALLOCATION DE RESSOURCES DANS LES CLOUD COMPUTING*, Oran - Algerie: Univesité ahmed ben bella, 2016, pp. 21-24.
- [17] T. J. V. R. E. Anthony T. Velte, *Cloud Computing: A Practical Approach*, London: The McGraw-Hill, 2010.
- [18] T. Aladwani, "Types of Task Scheduling algorithms in Cloud Computing," *Intechopen*, vol. 0, no. 0, pp. 1-13, 2020.
- [19] J. P. F. T. Frédéric Magoulès, *Cloud Computing: Data-Intensive Computing and Scheduling*, USA: CRC, 2013.
- [20] Y. Z. Wenhong Tian, *Optimized Cloud Resource Management and Scheduling*, USA: Elsevier, 2015.
- [21] B. B. Zouheyr Bouafia, "Algorithmes d'ordonnancement des tâches dans un environnement Cloud," *Revue Méditerranéenne des Télécommunications*, vol. 05, no. 02, pp. 71-78, 2015.

-
- [22] O. B. Younes Hajoui, "New Hybrid Task Scheduling Algorithm with Fuzzy Logic Controller in Grid computing," *International Journal of Advanced Computer Science and Applications*, vol. 09, no. 08, pp. 547-554, 2018.
- [23] G. J. T. Sudha, "Fuzzy Fractals in Cervical Cancer," *Fuzzy Intelligent Systems: Methodologies, Techniques, and Applications*, pp. 1-25, 2021.
- [24] H. D. V. S. Mojtaba Sabeghi, "A FUZZY ALGORITHM FOR REAL-TIME SCHEDULING," in *IADIS International Conference Applied Computing*, USA, 2006.
- [25] L.-X. Wang, *A course in fuzzy systems and control*, USA: Prentice Hall international, 1997.
- [26] A. E. O. N. A. Awatif RAGMANI, "An Improved Scheduling Strategy in Cloud Computing Using Fuzzy Logic," in *Proceedings of the International Conference on Big Data and Advanced Wireless Technologies*, Bulgaria, November 10 - 11, 2016.
- [27] M. O. M. H. Z. A. M. L. Ala'anzy, "Inspired Algorithm for Cloudlet Scheduling in Cloud Computing Environments," *Sensors*, p. 21, 2021.
- [28] Sourceforge, "www.sourceforge.net," 20 2 2020. [Online]. Available: <https://sourceforge.net/projects/jfuzzyqt/>. [Accessed 2022 6 4].
- [29] Y. Z. Y. D. a. S. F. Zhijia Chen, "A Dynamic Resource Scheduling Method Based on Fuzzy Control Theory in Cloud Environment," *Journal of Control Science and Engineering*, vol. 03, no. 10, pp. 1-10, 2015.