

الجمهورية الجزائرية الديمقراطية الشعبية
République algérienne démocratique et populaire
وزارة التعليم العالي والبحث العلمي
Ministère de l'enseignement supérieur et de la recherche scientifique
جامعة عين تموشنت بلحاج بوشعيب
Université –Ain Temouchent- Belhadj Bouchaib
Faculté des Sciences et de Technologie
Département Mathématique et Informatique



Projet de Fin d'Etudes
Pour l'obtention du diplôme de Master en Informatique
Domaine : Mathématiques et Informatique
Filière : Informatique
Spécialité : Cybersécurité et Intelligence artificielle.
Thème

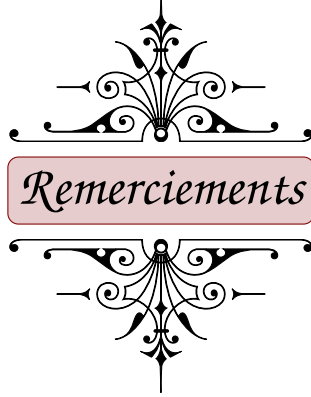
***Réduction de la consommation d'énergie
dans le Cloud Computing.***

Présenté par :
Benbouha Djihene
Hassini Imene

Devant le jury composé de :

Dr. Mededjel .M	UAT.B.B	Président
Dr. Merad Boudia .D	UAT.B.B	Examineur
Dr. Bouafia .Z	UAT.B.B	Encadrant

Année Universitaire : 2024-2025



Nous remercions tout d'abord ALLAH tout puissant pour nous avoir donné la santé, la patience et le courage pour mener à bien ce mémoire.

Nous remercions nos très chers parents qui ont toujours été là pour nous.

Nous remercions notre sœur et nos frères pour leurs encouragements.

Nous tenons à témoigner notre plus profonde gratitude à notre encadrant Monsieur BOUAFIA Zouheyr qui a dirigé ce travail, nous a encadrés et proposé un sujet aussi passionnant et intéressant. Sa disponibilité permanente et son aide nous ont été d'un soutien précieux dont nous lui sommes particulièrement reconnaissants. Sa compétence et ses conseils nous ont été d'un grand secours.

Encore merci!

Par ailleurs, nous tenons à exprimer nos vifs remerciements au président du jury Monsieur M'EDJEL Mansour et Monsieur Merad Boudia Djalal . Nous remercions très respectueusement qui ont consacré leur temps à examiner et juger notre travail avec attention.

D'autre part, nous adressons une chaleureuse pensée à toute l'équipe pédagogique du Département de Mathématiques et Informatique, en particulier à la Cheffe de département Madame Hellal Meriem.

Enfin, un grand merci à toute personne nous ayant aidés et guidés pour la réalisation de ce travail.





Dédicaces

A

Tous ceux que j'aime et qui m'aiment.

A

Mes Parents Mohamed et Fatima qui m'ont encouragée et soutenue

A

mes frères Abdelhadi et Abdelrahim et Ahmed Abdelnoure

A

Mes grand-pères Mohammed et Miloud et mes grand-mères Bekhita et Halima

A

Mes cousines Chaimae et Maroua et mes tantes qui m'ont soutenue

A

Tous les membres des deux familles BENBOUHA et Ahmed El-Mezouar

A

A ma chère binôme Imene Mes amies surtout Basma , Houda

Benbouha Djihene 





A

Ma chère mère, pour son amour, ses sacrifices et son soutien inestimable tout au long de mon parcours.

A

Ma sœur Nadine et mon frère Sofiane, pour leur affection, leurs encouragements constants et leur présence rassurante.

A

Ma grand-mère, dont les prières et la bienveillance m'accompagnent chaque jour.

A

Mon binôme Djihene, pour sa collaboration, son sérieux et les efforts partagés tout au long de ce projet.

A

Mes amies Basma et Houda, pour leur soutien moral, leur amitié sincère et leur présence dans les moments importants.



A

L'ensemble des enseignants et membres du corps pédagogique du département de Math et Informatique de l'université Belhadj Bouchaib d'Aïn Témouchent pour la qualité de l'enseignement dispensé tout au long de ma formation.

A

Moi-même, pour ma persévérance, les efforts fournis et ma capacité à surmonter les obstacles tout au long de ce parcours exigeant.

A

Toutes les personnes qui, de près ou de loin, ont contribué à la réalisation de ce projet.

Hassini Imene 

Résumé

Le cloud computing présente une technologie prometteuse qui facilite l'exécution des applications dans divers domaines. Il fournit des services flexibles et évolutifs à la demande des utilisateurs avec une meilleure qualité de service. Cependant, cette flexibilité engendre une forte consommation d'énergie au niveau des centres de données, véritables piliers de l'infrastructure cloud, dont les besoins énergétiques connaissent une croissance rapide. Face à ce enjeu environnemental et économique de haut niveau, la réduction de l'impact énergétique de cette infrastructure s'impose comme une priorité majeure.

Dans ce projet de fin d'étude, nous nous sommes intéressées à la réduction de la consommation d'énergie dans le cloud computing notamment en mettant en œuvre trois approches principales : mise à l'échelle dynamique de la tension et de la fréquence (DVFS) et la consolidation dynamique de machines virtuelles et aussi la combinaison de ces deux techniques. Nous avons simulé ces techniques à l'aide de la bibliothèque SimPy sous Python et analysé leur impact sur plusieurs indicateurs clés tels que la consommation d'énergie, le temps d'exécution et le respect de délai d'exécution.

Nos résultats montrent que DVFS est adapté pour réduire la consommation d'énergie (jusqu'à 52 %) alors que la consolidation dynamique permet d'optimiser la consommation d'énergie jusqu'à 22 %, tandis que la méthode combinée atteint une réduction de 49 %. Cette approche combinée a donné des performances importantes, notamment pour les charges de travail élevées.

Mots-clés : Cloud Computing, Consommation d'énergie, DVFS, Consolidation des machines virtuelles, Centre de données, SimPy, Qualité de service.

Abstract

Cloud computing is a promising technology that facilitates the execution of applications in various fields. It provides flexible and scalable services on demand with improved quality of service. However, this flexibility leads to high energy consumption in data centers, which are the backbone of cloud infrastructure and whose energy needs are growing rapidly. Faced with this significant environmental and economic challenge, reducing the energy impact of this infrastructure is a major priority.

In this final year project, we focused on reducing energy consumption in cloud computing, in particular by implementing three main approaches : dynamic voltage and frequency scaling (DVFS), dynamic consolidation of virtual machines, and a combination of these two techniques. We simulated these techniques using the SimPy library in Python and analyzed their impact on several key indicators such as energy consumption, execution time, and execution deadline compliance.

Our results show that DVFS is suitable for reducing energy consumption (up to 52%), while dynamic consolidation optimizes energy consumption by up to 22%, and the combined method achieves a reduction of 49%. This combined approach yielded significant performance gains, particularly for high workloads.

Keywords : Cloud Computing, Energy consumption, DVFS, Virtual Machine Consolidation, Datacenter, SimPy, Quality of Service.

ملخص

تعد الحوسبة السحابية تقنية واعدة تسهل تشغيل التطبيقات في مختلف المجالات. فهي توفر خدمات مرنة وقابلة للتطوير حسب طلب المستخدمين مع جودة خدمة أفضل. ومع ذلك، فإن هذه المرونة تؤدي إلى استهلاك كبير للطاقة في مراكز البيانات، التي تعد ركائز البنية التحتية السحابية، والتي تشهد احتياجاتها من الطاقة نمواً سريعاً. في مواجهة هذا التحدي البيئي والاقتصادي الكبير، أصبح الحد من الأثر الطاقي لهذه البنية التحتية أولوية قصوى.

في هذا المشروع النهائي للدراسة، ركزنا على تقليل استهلاك الطاقة في الحوسبة السحابية، لا سيما من خلال تنفيذ ثلاث طرق رئيسية: التحجيم الديناميكي للجهد والتردد (DVFS) والدمج الديناميكي للألات الافتراضية، وكذلك الجمع بين هاتين التقنيتين. قمنا بمحاكاة هذه التقنيات باستخدام مكتبة SimPy في Python وقمنا بتحليل تأثيرها على عدة مؤشرات رئيسية مثل استهلاك الطاقة ووقت التنفيذ والالتزام بموعد التنفيذ. تظهر نتائجنا أن DVFS مناسب لتقليل استهلاك الطاقة (حتى 52٪) بينما يتيح التوحيد الديناميكي تحسين استهلاك الطاقة بنسبة تصل إلى 22٪، في حين تحقق الطريقة المركبة تخفيضاً بنسبة 49٪. وقد حققت هذه الطريقة المركبة أداءً مهمًا، لا سيما بالنسبة لأحمال العمل العالية.

الكلمات المفتاحية: الحوسبة السحابية، استهلاك الطاقة، التحجيم الديناميكي للجهد والتردد، الدمج الديناميكي للألات الافتراضية، مركز البيانات، SimPy، جودة الخدمة.

Table des matières

Introduction générale	17
1 Introduction au cloud computing	19
1.1 Introduction	20
1.2 Historique	20
1.3 Définition	21
1.4 Notions de base	21
1.5 Caractéristiques du cloud computing	22
1.5.1 Ressources à la demande	22
1.5.2 Large accès réseau	23
1.5.3 Mutualisation de ressources (Pooling)	23
1.5.4 Élasticité rapide	23
1.5.5 Service mesurable et facturable (Pay As You Go)	23
1.6 Modèles de service	23
1.6.1 Infrastructure as a service (IaaS)	23
1.6.2 Plateforme as a service (PaaS)	24
1.6.3 Software as a service (SaaS)	24
1.7 Modèles de déploiement	24
1.7.1 Cloud public	24
1.7.2 Cloud privé	25
1.7.3 Cloud hybride	26
1.7.4 Cloud communautaire	26
1.8 Fournisseurs du cloud	27
1.9 Avantages de cloud computing	28
1.10 Problèmes de gestion de cloud computing	28
1.11 Conclusion	29
2 Techniques de réduction de la consommation d'énergie	30
2.1 Introduction	31
2.2 Énergie et puissance électrique	31
2.3 Paramètres d'efficacité énergétique	32
2.3.1 Power Usage Effectivness (PUE)	32

2.3.2	Energy Reuse Effectiveness (ERE)	32
2.3.3	Data Center Infrastructure Efficiency (DCiE)	33
2.4	Consommation d'énergie dans le cloud computing	33
2.4.1	Répartition de consommation énergétique d'un centre de données	34
2.5	Techniques de réduction de la consommation d'énergie	34
2.5.1	Gestion statique de l'énergie	35
2.5.1.1	Utilisation de matériel informatique écoénergétique	35
2.5.1.2	Optimisation énergétique du CPU et des systèmes	35
2.5.1.3	C-states	36
2.5.1.4	Optimisation du refroidissement des centres de données	36
2.5.1.5	Consolidation statique de machines virtuelles	37
2.5.2	Gestion dynamique de l'énergie	37
2.5.2.1	Mise à l'échelle dynamique de la fréquence et de la tension (DVFS)	38
2.5.2.2	Ordonnancement des tâches	38
2.5.2.3	Consolidation dynamique de machines virtuelles	39
2.5.2.4	Power Capping	40
2.5.2.5	Approvisionnement économe en énergie	41
2.6	Quelques travaux de recherche sur l'optimisation énergétique des centres de données	41
2.6.1	Simulateur de mise à l'échelle dynamique de la tension et de la fréquence pour la gestion énergétique des flux de travail réels dans le cloud computing vert	41
2.6.2	Consolidation auto-adaptative des machines virtuelles pour l'efficacité énergétique dans le cloud	42
2.6.3	Ordonnancement des tâches bi-objectifs axé sur la performance et la consommation d'énergie pour les centres de données cloud	43
2.6.4	Ordonnancement des tâches écoénergétique basé sur le DVFS dans un centre de données.	44
2.6.5	Technologies de free cooling pour les centres de données : mécanisme d'économie d'énergie et applications	45
2.6.6	Comparaison des travaux de recherche de réduction de la consommation énergétique dans le cloud computing	45
2.7	Conclusion	48
3	Implémentation et Simulations	49
3.1	Introduction	50
3.2	Outils et l'environnement de développement	50
3.2.1	Langage de programmation Python	50

3.2.2	Visual Studio Code	51
3.2.3	Bibliothèques Python utilisées	51
3.2.4	Paramètres de l'environnement	52
3.3	Calcul de l'énergie consommée	53
3.4	Approches utilisés pour l'optimisation énergétique	54
3.4.1	Mise à l'échelle dynamique de la tension et de la fréquence (DVFS)	54
3.4.2	Consolidation dynamique des VMs	55
3.4.3	Combinaison des techniques DVFS et de consolidation	56
3.5	Pseudocodes des mécanismes d'optimisation énergétique	56
3.5.1	Pseudocode de l'algorithme DVFS	56
3.5.2	Pseudocode de l'algorithme de consolidation	57
3.5.3	Pseudocode de l'algorithme combiné DVFS et consolidation	59
3.6	Résultats et analyses	60
3.6.1	Résultats de simulation pour 500 tâches	60
3.6.2	Résultats de simulation pour 1000 tâches	60
3.6.3	Résultats de simulation pour 1500 tâches	61
3.6.4	Résultats de simulation pour 2000 tâches	62
3.6.5	Consommation d'énergie par rapport au nombre de tâches	62
3.6.6	Temps d'exécution par rapport au nombre des tâches	63
3.6.7	Respect de délai par rapport au nombre de tâches	64
3.7	Synthèse	65
3.8	Conclusion	66
	Conclusion générale	68
	Bibliographie	70

Table des figures

1.1	Cloud Computing.	21
1.2	Services du cloud computing.	24
1.3	Modèle de déploiement d'un cloud public.	25
1.4	Modèle de déploiement d'un cloud privé.	25
1.5	Modèle de déploiement d'un cloud hybride.	26
1.6	Modèle de déploiement d'un cloud communautaire.	26
2.1	Répartition de consommation énergétique d'un centre de données.	34
2.2	Répartition des techniques de gestion de l'énergie.	35
2.3	Ordonnancement des tâches.	39
3.1	Consommation d'énergie par rapport au nombre de tâches.	63
3.2	Temps d'exécution par rapport au nombre des tâches.	64
3.3	Respect de délai par rapport au nombre de tâches.	65

Liste des tableaux

2.1	Comparaison des travaux de recherche de réduction de la consommation énergétique dans le cloud computing.	48
3.1	Paramètres des machines physiques	52
3.2	Paramètres des machines virtuelles	52
3.3	Paramètres des tâches	52
3.4	Les paramètres de base	53
3.5	Résultats pour 500 tâches.	60
3.6	Résultats pour 1000 tâches.	61
3.7	Résultats pour 1500 tâches.	61
3.8	Résultats pour 2000 tâches.	62

Liste des abréviations

ACS	Ant Colony System.
API	Application Programming Interface.
AWS	Amazon Web Service.
CUE	Carbon Usage Effectiveness.
CPU	Central Processing Unit.
DPC	Decentralized Power Capping.
DRS	Distributed Resource Scheduler.
DVFS	Dynamic Voltage Frequency Scaling.
ECR	Energy Consumption Ratio.
ERE	Energy Reuse Effectiveness.
ERF	Energy Reuse Factor.
ETSI	European Telecommunications Standards Institute.
FAWN	Fast Array of Wimpy Nodes.
FFD	First Fit Decreasing.
GCP	Google Cloud Platform.
HDD	Hard Disk Drive.
IA	Intelligence Artificielle.
IaaS	Infrastructure as a Service.
IBM	International Business Machines.
IDE	Integrated Development Environment.
IT	Information Technology.
MI	Million Instructions.
MBFD	Modified Best Fit Decreasing.
MIPS	Million Instructions Per Second.
MIT	Massachusetts Institute of Technology.

MWFD	Modified Worst Fit Decreasing.
NIST	National Institute of Standards and Technology.
OS	Operating System.
PaaS	Platform as a Service.
PM	Physical Machine.
PTAB	Power-Aware Task Allocation for Batch Tasks.
PUE	Power Usage Effectiveness.
QoS	Quality of Service.
RAM	Random-Access Memory.
SaaS	Software as a Service.
SSD	Solid State Drive.
VPN	Virtual Private Network.
WFD	Worst Fit Decreasing.

Introduction générale

Introduction générale

Dans un contexte mondial marqué par une transition numérique accélérée et une sensibilisation croissante aux enjeux environnementaux, le cloud computing devient la technologie de référence. Ce modèle amène à disposer d'accès, de traitements et de stockage de données mobilisant sur Internet des ressources informatiques sur demande, avec une certaine souplesse, à des coûts moindres. Nos questionnements quant à cette promesse d'optimisation et d'efficacité s'expliquent aujourd'hui par un problème grandissant, l'élément clé de l'environnement cloud, à savoir les centres de données, se montrant de plus en plus énergivores. En effet, les centres de données, fonctionnant continuellement et du fait de l'augmentation du nombre de services hébergés, représentent une fraction importante de la consommation énergétique mondiale. Cette surconsommation génère des coûts élevés, mais également une empreinte carbone importante. La recherche de solutions pour réduire la consommation énergétique dans le cloud computing devient une préoccupation majeure, tant pour les chercheurs que pour les fournisseurs de services cloud.

L'objectif principal de ce projet de fin d'études consiste donc à présenter et à évaluer des techniques de réduction de la consommation d'énergie, adaptées à l'application au cloud computing, comme le DVFS ou la consolidation dynamique de machines virtuelles, afin de réduire de manière significative la consommation d'énergie des centres de données tout en tentant de conserver un niveau de qualité de service adéquat. Le rapport est organisé comme suit :

Le premier chapitre traite des fondamentaux du cloud computing, en abordant dans un premier temps son historique, ses définitions, puis les concepts essentiels comme la virtualisation et les modèles de services et de déploiement, mais également les caractéristiques du cloud, ses avantages, ainsi que les difficultés de sa gestion dont la consommation énergétique.

Dans le deuxième chapitre, plusieurs techniques de réduction de la consommation d'énergie dans les centres de données sont décrites, les paramètres d'efficacité énergétique sont listés, ainsi que des sources principales de consommation, puis il distingue les approches statiques, comme par exemple les matériels écoénergétiques, et les C-states, ou les approches dynamiques à travers le DVFS, la consolidation de machines virtuelles, ou encore le power capping. ... Tout en analysant des travaux de recherche récents concernant l'optimisation énergétique dans le cloud.

Le dernier chapitre est consacré à la description de l'environnement de développement, des outils utilisés ainsi que de la mise en œuvre des mécanismes. Trois algorithmes de réduction de la consommation énergétique ont été simulés, mise à l'échelle dynamique de la tension et de la fréquence, ainsi que la consolidation dynamique des machines virtuelles. Une combinaison sous forme de coordination entre les deux techniques a aussi été testée afin de quantifier leur efficacité combinée. Les résultats de simulation sont ensuite analysés selon plusieurs critères : consommation d'énergie, temps d'exécution, respect des délais d'exécution.

Nous terminons ce rapport par une conclusion générale qui synthétise les techniques et les résultats obtenus en matière de réduction de la consommation énergétique dans le cloud. Elle met en lumière l'importance de ces techniques de réduction d'énergie et conduit à approfondir les travaux futurs autour de méthodes d'optimisation plus intelligentes.

CHAPITRE 1

INTRODUCTION AU CLOUD COMPUTING

1.1 Introduction

Dans le but de simplifier la gestion des ressources informatiques et de réduire la dépendance aux équipements physiques, le cloud computing a été conçu. Il s'impose comme une révolution majeure, transformant la gestion et l'exploitation des ressources informatiques. Cette innovation permet aux organisations et aux particuliers d'accéder à des services informatiques à distance, sans avoir à investir dans des infrastructures coûteuses. L'essor du cloud computing est lié à la nécessité croissante de flexibilité, de performance et d'optimisation des coûts. En offrant un accès à la demande à des ressources de calcul, de stockage et d'applications, il répond aux besoins variés des entreprises, allant des startups aux grandes multinationales. Grâce à des modèles économiques basés sur la facturation à l'usage, il facilite l'adoption des technologies avancées sans nécessiter un investissement initial important. Dans ce chapitre, nous allons présenter quelques généralités sur le cloud computing, à savoir sa définition, ses caractéristiques, les différents modèles de services et de déploiement. Le chapitre présente également les principaux fournisseurs de services cloud, les avantages de cette technologie, ainsi les défis liés à cette technologie.

1.2 Historique

Le concept de cloud computing a été introduit pour la première fois dans une communication scientifique en 1961, par le pionnier du calcul, John McCarthy, dans le cadre d'une conférence tenue au MIT. C'est en 1999 que le premier hébergeur de cloud, Salesforce.com, devint opérationnel. En 2002, Amazon proposa des services d'hébergement, puis, en 2005, fut mis en place Amazon Web Services (AWS), avant que Google, IBM puis Microsoft, ne cherchent enfin à proposer eux aussi leurs propres services de cloud computing, à partir de 2009.

Les années 2010 sont marquées par l'essor du cloud avec notamment le lancement de Microsoft Azure (2010), OpenStack (2010), IBM SmartCloud (2011) ou encore iCloud (2011). Google Compute Engine et Oracle Cloud sont lancés en 2012, puis Docker en 2013 révolutionnant le développement avec les conteneurs. En 2019, Amazon a lancé AWS Outposts, un service qui étend l'infrastructure, les services, les API et les outils d'AWS aux centres de données des clients, aux espaces de co-implantation ou aux installations sur site. Depuis la pandémie mondiale de 2020, la technologie du cloud a connu un essor en popularité grâce au niveau de sécurité des données et à la flexibilité des options de travail pour tous les employés, notamment les travailleurs à distance.

[7]

1.3 Définition

Le cloud computing constitue un modèle innovant où des fournisseurs de services offrent des ressources informatiques flexibles, fréquemment virtualisées, et disponibles de façon dynamique. Cette technologie permet aux utilisateurs d'accéder, via Internet, à divers services tels que des applications, du stockage ou des plateformes de développement, à partir d'appareils comme des ordinateurs, des smartphones ou des assistants numériques.

Le cloud computing est formellement défini par l'institut national des normes et de la technologie (NIST) comme un modèle permettant un accès réseau pratique et omniprésent à la demande à un pool partagé de ressources informatiques configurables (réseaux, serveurs, stockage, applications et services). Ces ressources pouvant être rapidement réservées, utilisées et libérées avec un effort minimal de gestion et d'interaction du fournisseur de services.[10]

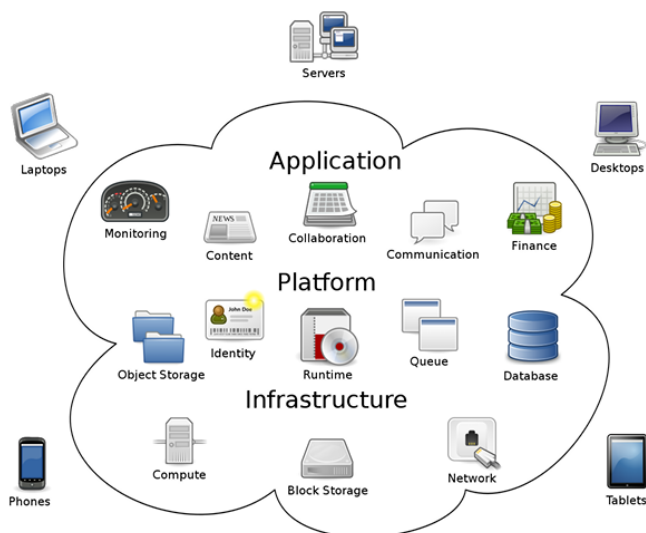


FIGURE 1.1 – Cloud Computing.[36]

1.4 Notions de base

-Centre de données (data center) : est une infrastructure composée d'un réseau d'ordinateurs et d'espaces de stockage. Il peut être utilisé par les entreprises pour organiser, traiter, stocker de grandes quantités de données. Cette infrastructure comprend plusieurs éléments :

- Salles sécurisées destinées à accueillir les équipements informatiques : armoires de raccordement pour les serveurs aux dimensions standardisées, serveurs applicatifs exécutant les logiciels, et serveurs de données assurant le stockage des informations, équipements réseau, permettant l'interconnexion des serveurs, tels que des routeurs, pare-feu, répartiteurs et commutateurs.

- Infrastructures techniques garantissant la continuité de l'alimentation électrique.
- Refroidissement des serveurs.
- Points d'accès aux réseaux électriques à haute tension et aux réseaux de télécommunication très haut débit.
- Bâtiment spécialisé et sécurisé intégrant l'ensemble de ces composants. [15]

-Machine Virtuelle : Une machine virtuelle (VM) est une imitation logicielle de matériel physique qui permet à un système d'exploitation et à des applications de s'exécuter comme s'ils se trouvaient sur des machines physiques distinctes, bien qu'ils s'exécutent en réalité sur le même matériel sous-jacent. Les machines virtuelles fonctionnent comme des machines indépendantes créées grâce à une technologie appelée virtualisation, qui permet à plusieurs instances virtuelles de s'exécuter sur un seul serveur physique.[49]

-Machine Physique : Dans le domaine de l'informatique, une machine physique est un serveur ou un ordinateur dont les ressources matérielles (processeur, mémoire, stockage, etc.) sont dédiées à exécuter directement un système d'exploitation et des applications, sans virtualisation ni partitionnement matériel. À la différence de la machine virtuelle, la machine physique repose entièrement sur son matériel (non partagé) et n'est pas encapsulée dans un hyperviseur.[1]

-Broker : Broker est une entité intermédiaire qui gère l'usage, la performance et l'approvisionnement des services cloud en facilitant la relation entre les fournisseurs (cloud providers) et les utilisateurs (cloud consumers). Il négocie l'accès aux ressources cloud et peut offrir des services supplémentaires qui simplifient la gestion pour les consommateurs. Son rôle inclut l'orchestration, l'agrégation et l'arbitrage des services cloud, permettant ainsi une optimisation des coûts, une meilleure intégration des solutions et une amélioration des performances pour les entreprises.[35]

-Virtualisation : Est une technologie permettant d'optimiser l'utilisation des ressources matérielles en créant plusieurs machines virtuelles sur une même machine physique. Avec la virtualisation, chaque VM fonctionne comme un ordinateur autonome avec son propre système d'exploitation, tout en partageant les ressources matérielles sous-jacentes. Elle constitue un pilier fondamental du cloud computing. [48]

1.5 Caractéristiques du cloud computing

1.5.1 Ressources à la demande

Un utilisateur peut allouer des ressources informatiques (serveurs, réseau, stockage, environnement d'exécution, application) au besoin, de façon automatique et sans nécessité d'interaction humaine avec chaque fournisseur de services .[24]

1.5.2 Large accès réseau

Les ressources du cloud computing sont disponibles à travers le réseau et accessibles via des mécanismes standards qui favorisent leurs utilisations à partir des appareils clients hétérogènes, voire légères (ex ordinateurs portables, téléphones, tablettes).[24]

1.5.3 Mutualisation de ressources (Pooling)

Les ressources du cloud peuvent être regroupées pour servir des utilisateurs multiples, pour lesquels des ressources physiques et virtuelles sont automatiquement attribuées. En général, les utilisateurs n'ont aucun contrôle ou connaissance sur l'emplacement exact des ressources fournies. Toutefois, ils peuvent imposer de spécifier l'emplacement à un niveau d'abstraction plus haut.[24]

1.5.4 Élasticité rapide

Les ressources sont idéalement allouées et libérées automatiquement pour s'adapter rapidement à la demande, qu'elle soit croissante ou décroissante. Pour le consommateur, les ressources disponibles à l'allocation apparaissent comme illimitées et peuvent s'allouer à tout moment.[24]

1.5.5 Service mesurable et facturable (Pay As You Go)

Dans un environnement cloud, il est possible de mesurer de façon précise la consommation des différentes ressources (CPU, stockage, bande passante, etc). Cette mesure lui permet ensuite de facturer le client selon l'usage. Pay As You Go est une option de facturation typique pour un fournisseur de cloud computing, ce qui signifie que le client est facturé en fonction des ressources consommées. Cela peut être très utile dans le cas où ce dernier n'est pas certain des ressources dont il a besoin.[24]

1.6 Modèles de service

Il existe plusieurs types de services cloud, cependant dans ce mémoire, nous nous concentrerons uniquement sur les principaux :

1.6.1 Infrastructure as a service (IaaS)

Infrastructure en tant que service offre des ressources d'infrastructure virtualisées, telles que la puissance de calcul, le stockage et le réseau, sous forme de services publics. Il libère les utilisateurs de la gestion du matériel physique, en leur permettant d'accéder à des machines virtuelles (MVs). Ces utilisateurs ont un contrôle quasi complet sur leurs MVs, ils sont responsables de la gestion du système d'exploitation, des applications et des données déployées, ils peuvent également gérer l'accès aux machines via différents ports Internet. L'IaaS permet d'héberger des logiciels, des

applications et de stocker des données, avec une facturation basée sur l'utilisation des ressources. [12]

1.6.2 Plateforme as a service (PaaS)

Plateforme en tant que service est un modèle cloud dédié au développement d'applications. Il fournit une plateforme entièrement configurée et gérée, permettant à l'utilisateur de développer, tester et déployer ses applications sans avoir à acheter de logiciels ni effectuer d'installations supplémentaires. Le déploiement des solutions PaaS est automatisé, ce qui simplifie la gestion. Ce modèle offre également une grande flexibilité, idéale pour tester rapidement un prototype ou pour fournir un service informatique sur une période courte.[2]

1.6.3 Software as a service (SaaS)

Logiciel en tant que service offre aux utilisateurs des applications sous forme de services en ligne déjà déployés dans le cloud. Il offre aux clients la possibilité de consommer ces applications à la demande via l'internet. Ces applications, prêts à l'emploi, et ne nécessitent ni maintenance, ni installation de logiciel, ni mise à jour. Toutes ces opérations sont effectuées par le fournisseur d'application, dans SaaS l'utilisation d'application reste transparente pour les utilisateurs, qui ne se soucient ni de la plateforme, ni du matériel .[7]

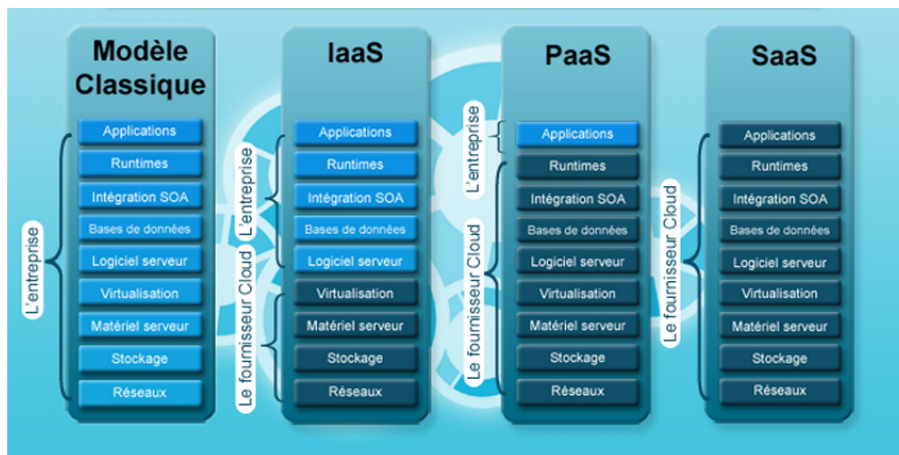


FIGURE 1.2 – Services du cloud computing.[16]

1.7 Modèles de déploiement

1.7.1 Cloud public

Le cloud public est un modèle largement utilisé où les ressources informatiques sont mises à disposition par des fournisseurs tiers via Internet. Ces ressources, hébergées dans un ou plusieurs centres

de données, sont partagées entre plusieurs clients et auto-provisionnées dynamiquement à travers des applications ou services web. Les fournisseurs assurent la supervision, la sécurisation et la gestion des infrastructures matérielles et logicielles, limitant ainsi le contrôle des utilisateurs sur la configuration de leurs ressources. La facturation est basée sur l'utilisation des services, offrant une accessibilité à tout client souhaitant exploiter ces ressources cloud.[7]

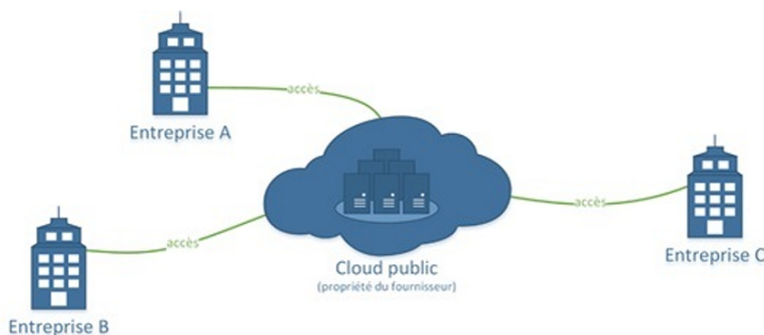


FIGURE 1.3 – Modèle de déploiement d'un cloud public. [7]

1.7.2 Cloud privé

L'ensemble des ressources d'un cloud privé est exclusivement mis à disposition d'une entreprise ou organisation unique. Le cloud privé peut être géré par l'entreprise elle-même (cloud privé interne) ou par une tierce partie (cloud privé externe). Les ressources sont généralement situées dans les locaux de l'entreprise ou chez un fournisseur de services, mais dans ce dernier cas, l'infrastructure reste entièrement dédiée à l'entreprise et est accessible via un réseau sécurisé, tel qu'un VPN. L'utilisation d'un cloud privé garantit que les ressources matérielles allouées ne seront jamais partagées entre plusieurs clients.[2]

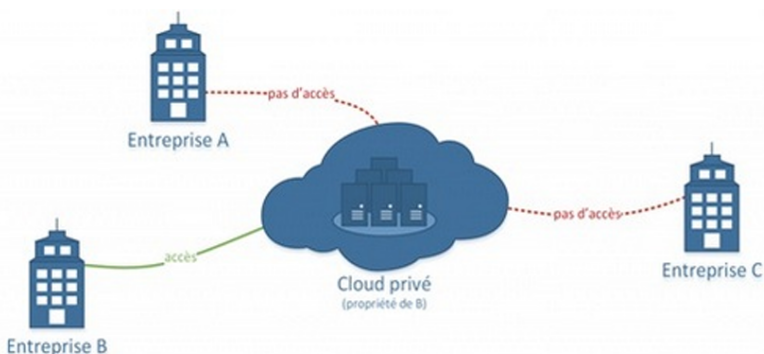


FIGURE 1.4 – Modèle de déploiement d'un cloud privé.[7]

1.7.3 Cloud hybride

Un cloud hybride combine les deux modes de déploiement précédents, cherchant à équilibrer leurs avantages et inconvénients. Dans ce modèle, les ressources sensibles sont déployées dans un cloud privé, tandis que les autres sont placées dans un cloud public. Les organisations optent généralement pour ce modèle afin de réduire les coûts tout en maintenant un niveau de sécurité acceptable. Bien que le cloud hybride offre une plus grande flexibilité, sa conception nécessite une planification minutieuse pour déterminer la répartition optimale des ressources et gérer leur circulation entre les deux cloud.[13]

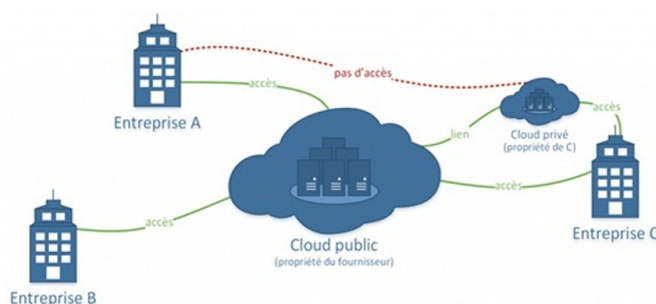


FIGURE 1.5 – Modèle de déploiement d'un cloud hybride. [7]

1.7.4 Cloud communautaire

Le cloud communautaire est un modèle de déploiement partagé entre plusieurs entreprises ou organisations, géré et sécurisé soit par les participants eux-mêmes, soit par un fournisseur de services. Il s'agit d'une forme hybride de cloud privé, conçu spécifiquement pour un groupe ayant des exigences communes. Ces communautés, comme des organismes gouvernementaux ou des entreprises de télécommunication, unissent leurs ressources humaines et financières pour atteindre des objectifs partagés, notamment en matière de réseau, de sécurité, de stockage et de calcul.[7]

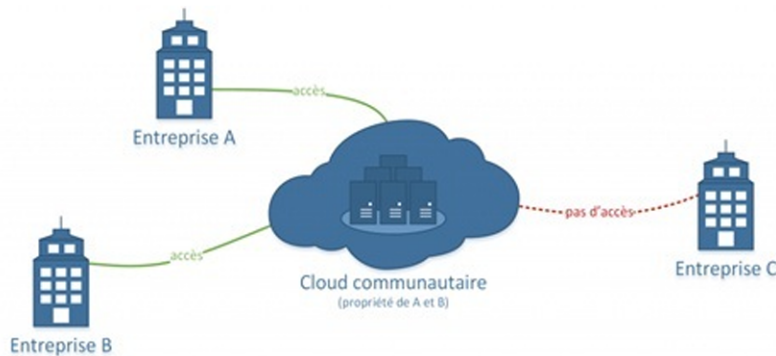


FIGURE 1.6 – Modèle de déploiement d'un cloud communautaire. [7]

1.8 Fournisseurs du cloud

À l'arrivée du cloud, qui représente une révolution dans le monde informatique, de nombreuses solutions ont été proposées, offrant divers avantages. Aujourd'hui, plusieurs fournisseurs proposent différents types de services cloud, tels que des applications, des outils, des environnements de programmation ou de développement, ainsi que des ressources de stockage et de calcul hébergées dans des centres de données. Dans cette section, nous présenterons les principaux fournisseurs, cités comme suit :

Amazon Web Services (AWS) : Lancé en 2006, AWS est le plus grand fournisseur mondial de services cloud, offrant des solutions d'infrastructure (IaaS), de plateforme (PaaS) et de logiciels (SaaS) adaptées aux entreprises de toutes tailles. Sa large gamme de services inclut le calcul (EC2), le stockage (S3), les bases de données (RDS, DynamoDB) et l'intelligence artificielle (SageMaker). AWS se distingue également par ses solutions avancées de sécurité (IAM, KMS) et son écosystème évolutif, permettant aux entreprises d'innover avec agilité.[39]

Microsoft Azure : Principal concurrent d'AWS, propose une large gamme de services adaptés aux besoins de diverses entreprises. Ce qui distingue Azure, c'est son intégration étroite avec l'écosystème Microsoft (Office 365, Windows Server) et ses solutions de cloud hybride. La plateforme offre des services d'infrastructure (IaaS), de développement (PaaS) et de gestion des bases de données, ainsi que des outils avancés d'IA et de sécurité, notamment Azure Active Directory.[39]

Google Cloud Platform (GCP) : Lancé en 2008, GCP est la solution cloud de Google, reconnue pour ses services avancés en big data et intelligence artificielle. Grâce à des outils tels que Google Compute Engine et BigQuery, GCP se distingue dans l'analyse de grandes quantités de données et le machine learning. Sa spécialisation en open source et son infrastructure mondiale assurent des performances optimales et une interopérabilité élevée.[39]

Oracle Cloud : Oracle Cloud est spécialement optimisé pour les bases de données et les applications d'entreprise, offrant une migration fluide et une gestion simplifiée des solutions Oracle existantes. Il permet aux entreprises utilisant déjà les produits Oracle de profiter de l'élasticité du cloud et de réduire leurs coûts. Son point fort réside dans ses bases de données autonomes, qui assurent une gestion plus efficace et sécurisée grâce à l'automatisation et à l'intelligence artificielle.[38]

IBM Cloud : IBM Cloud se distingue par son expertise en cloud hybride et en sécurité, répondant aux besoins des grandes entreprises et des secteurs réglementés comme la finance et la santé. Il intègre IBM Watson, offrant des capacités avancées d'intelligence artificielle et d'analyse de données. IBM Cloud se démarque également par ses solutions blockchain, optimisant la sécurité et la transparence des transactions ainsi que la gestion des chaînes d'approvisionnement.[38]

1.9 Avantages de cloud computing

Utilisation simplifiée : L'informatique en nuage augmente la facilité d'utilisation en surmontant les contraintes informatiques traditionnelles, telles que l'installation et la maintenance des logiciels, le stockage des données et la portabilité. Il y a une plus grande flexibilité et agilité car l'accès aux ressources informatiques peut être réalisé via un portail web, éliminant ainsi le besoin de matériel supplémentaire.[6]

Diminution des pertes de données accidentelles : La sauvegarde en temps réel de toute information majeure de l'institut, accompagnée d'une précision constante, assure que nos utilisateurs peuvent consommer les informations pertinentes qu'ils souhaitent, lorsqu'ils le désirent.[25]

Favoriser la collaboration en équipe : Le cloud offre une grande opportunité pour de nombreux utilisateurs car il permet à plusieurs utilisateurs de travailler facilement ensemble sur des documents et des projets. [5]

Portabilité : Les organisations ont la possibilité d'exploiter leur capacité de calcul n'importe où, accessible par les utilisateurs à diverses localisations géographiques. [15]

Optimisation des coûts : Les services de cloud offrent à une entreprise la possibilité de lancer une activité professionnelle sans avoir à déboursier pour l'infrastructure informatique, qui implique aussi une gestion interne plus onéreuse comprenant l'acquisition de matériel, l'attribution du personnel, la location de locaux, la consommation d'électricité, l'entretien et le renouvellement du matériel. Un autre aspect intéressant est le système de facturation à l'utilisation, ce qui implique que l'utilisateur ne paie que pour ce qu'il consomme réellement.[25]

1.10 Problemes de gestion de cloud computing

Malgré les nombreux avantages du cloud computing, plusieurs problèmes persistent et de nouveaux défis continuent d'apparaître. Dans cette section, nous présentons un résumé des principaux enjeux et problèmes liés au cloud computing :

Migration des données entre les environnements non standards : La majorité des fournisseurs de services cloud exploitent des applications propriétaires, limitant leur interopérabilité. Cette absence de compatibilité complique le transfert des données vers un autre fournisseur de cloud ou leur rapatriement sur des infrastructures locales.[12]

Sécurité de données et confidentialité : Dans le cloud computing, les données transitent entre les dispositifs des utilisateurs et les centres de données des fournisseurs de services, les exposant ainsi aux cyberattaques. Il est donc essentiel de garantir leur sécurité et leur confidentialité, aussi bien lors du transfert sur le réseau que durant leur stockage dans les centres de données du cloud.[12]

Problèmes de performance : La qualité des applications cloud dépend de la vitesse et de la

stabilité de la connexion Internet. Une latence élevée peut ralentir l'accès aux services et nuire à la productivité.[12]

Gestion de l'énergie : L'optimisation énergétique est un enjeu central du cloud computing qui fait face à une consommation largement croissante, en effet en 2006, les centres de données qui ont été recensés aux États-Unis représentaient plus de 1,5 % de la consommation énergétique nationale, un chiffre qui a bondi de 18 % par an et qui a grossi de près de 4,4 % de l'électricité totale du pays au début de 2023. Les besoins en électricité des centres de données est ainsi passé de 58 TWh en 2014 à 176 TWh en 2023. Face à cette montée continue, les fournisseurs doivent mettre en place des méthodes adéquates pour réduire la consommation d'énergie tout en maintenant un haut niveau d'efficacité.[40]

1.11 Conclusion

Le cloud computing représente une transformation majeure de l'informatique en proposant des services flexibles, accessibles et évolutifs. Il permet aux entreprises et aux utilisateurs d'optimiser leurs ressources tout en réduisant les coûts. Son adoption continue de croître grâce aux avancées technologiques et à ses nombreux avantages. Cependant, cette évolution s'accompagne d'une consommation croissante d'énergie en rapport avec le fonctionnement de l'infrastructure cloud. Dans ce chapitre, nous avons exploré les notions fondamentales du cloud computing, en abordant ses modèles de déploiement, ses services, ainsi que ses caractéristiques, avantages et inconvénients. Dans le chapitre suivant, nous allons présenter les méthodes fondamentales permettant de minimiser la consommation d'énergie, en détaillant leurs principes et leurs implications, afin d'offrir une vision globale des stratégies adoptées pour améliorer la gestion énergétique dans ce domaine.

CHAPITRE 2

TECHNIQUES DE RÉDUCTION DE LA CONSOMMATION D'ÉNERGIE

2.1 Introduction

L'essor et l'utilisation des services d'application en cloud ont conduit à une consommation croissante de ressources informatiques, et par conséquent à une consommation croissante d'énergie par les centres de données. Ces infrastructures essentielles à la fourniture de services cloud, doivent être gérées de façon appropriée afin de réduire les coûts liés à l'utilisation de ces ressources, tout en limitant l'impact environnemental généré par l'augmentation constante de leur consommation énergétique. L'optimisation de l'énergie est un sujet d'actualité pour les chercheurs et notamment pour les fournisseurs de services en cloud. Différentes stratégies ont été mises en œuvre pour réduire la consommation énergétique : optimisation du matériel et des infrastructures, mais aussi approches logicielles avancées comme la gestion dynamique des ressources, l'organisation et la gestion des tâches.

Ce chapitre présente plusieurs méthodes d'optimisation énergétique mises en œuvre dans les centres de données, telles que la consolidation dynamique des VMs, l'optimisation du refroidissement des infrastructures ou encore l'utilisation de matériel écoénergétiques, etc. Il se conclut par un tableau comparatif des travaux de recherche sur la réduction de la consommation d'énergie dans les centres de données. Ces travaux sont analysés selon leurs objectifs, méthodologies, avantages et inconvénients et métriques d'évaluation.

2.2 Énergie et puissance électrique

Pour comprendre les mécanismes de la gestion de l'énergie et de puissance, il est nécessaire de clarifier la terminologie. La puissance est la vitesse à laquelle un système exécute des tâches, tandis que l'énergie représente la quantité totale de travail effectué sur une période de temps. La puissance et l'énergie sont mesurées respectivement en watts (W) et en wattheures (Wh).

Formellement, la puissance et l'énergie peuvent être définis comme indiqué ci-dessous :

$$P = \frac{E}{T} \quad (1)$$

$$E = P \times T \quad (2)$$

Où P est la puissance électrique, T est une période de temps et E est l'énergie consommée.

La distinction entre puissance et énergie électrique est importante car réduire la puissance ne réduit toujours pas l'énergie consommée. Par exemple, la consommation d'énergie peut être réduite en réduisant les performances du processeur, auquel cas un programme ou une tâche peut mettre plus de temps à s'exécuter tout en consommant la même quantité d'énergie.

D'une part, la réduction de la puissance électrique de pointe entraîne une baisse des coûts d'acquisition d'infrastructures, tels que ceux liés à la capacité des alimentations sans interruption (UPS),

des unités de distribution d'énergie (PDU), des générateurs, des systèmes de refroidissement et des équipements de distribution d'énergie. D'un autre côté, une moindre consommation d'énergie peut réduire les factures d'électricité. [19]

2.3 Paramètres d'efficacité énergétique

Les paramètres d'efficacité énergétique dans les centres de données du cloud reposent sur plusieurs aspects essentiels visant à réduire la consommation d'énergie tout en maintenant des performances optimales.

2.3.1 Power Usage Effectivness (PUE)

L'indicateur d'efficacité énergétique PUE est un paramètre qui permet d'estimer l'efficacité d'un centre de donnée. Cet indicateur est défini par une association de professionnels issus du monde de la technologie d'information (Green Grid), elle élabore des standards et des méthodes pour augmenter efficacité des centres de données.

$$PUE = \frac{P_{total}}{P_{it}} \quad (3)$$

Le PUE correspond au rapport entre l'énergie totale consommée par le centre de données et l'énergie consommée par les équipements de la technologie de l'information IT (serveurs de calculs, stockage, équipements réseaux, etc.). Une valeur de 1 du PUE est considérée comme étant idéale, car elle signifie que toute la puissance du centre de données est consommée par les équipements informatiques.

En réalité cet indice est souvent supérieur à 2 pour la majorité des centres de données avec une valeur moyenne de 2,13. Quelques centres de données semblent avoir réussi à obtenir des PUE inférieur à 1,5, par exemple Google a annoncé un PUE moyen annuel de ses centres de données de 1,15. Mais le PUE reste critiqué par une partie de la communauté, car il est insuffisant pour comparer des centres de données pour plusieurs raisons :

- Un centre de données pourrait avoir un PUE faible malgré une consommation importante.
- La consommation des ventilateurs intégrés dans les serveurs n'est pas prise en compte.
- L'indicateur ne donne aucune information quant au service rendu.[18]

2.3.2 Energy Reuse Effectiveness (ERE)

L'ERE est une métrique qui évalue l'efficacité d'un centre de données à réutiliser son énergie excédentaire.Elle se calcule avec la formule :

$$ERE = (1 - ERF) \times PUE \quad (4)$$

où ERF (Energy Reuse Factor) représente la part d'énergie réutilisée à l'extérieur du centre (chauffage, industrie, etc.). Plus l'ERF est élevé, plus l'ERE est faible, ce qui signifie une meilleure efficacité. Un ERE bas indique donc une forte capacité à recycler l'énergie, réduisant l'impact environnemental et la dépendance aux énergies externes.

L'ERE complète les métriques comme la PUE (Power Usage Effectivness) et la CUE (Carbon Usage Effectiveness) en intégrant la notion de valorisation énergétique. Elle incite à des stratégies durables, notamment dans les régions froides où la chaleur résiduelle peut être réutilisée efficacement. Cela fait de l'ERE un indicateur clé dans les démarches de transition énergétique des centres de données.[26]

2.3.3 Data Center Infrastructure Efficiency (DCiE)

DCiE est une métrique qui permet de mesurer l'efficacité énergétique d'un centre de données.Elle se calcule ainsi :

$$\text{DCiE} = \frac{\text{Puissance des équipements IT}}{\text{Puissance totale de l'infrastructure}} \times 100 \quad (5)$$

Il s'agit du pourcentage d'énergie réellement fournie à l'ensemble des équipements informatiques par rapport à celle réellement consommée par l'ensemble du centre de données. Plus il est élevé, plus le centre est efficace. Ainsi, une DCiE de 50 % signifie que la moitié de l'énergie totale est effectivement allouée à la prise en charge de la logique informatique. Elle est l'inverse du PUE (Power Usage Effectiveness) et permet d'analyser les traitements des systèmes de support tels que le refroidissement ou l'alimentation pour identifier les sources de pertes.[31]

2.4 Consommation d'énergie dans le cloud computing

La demande croissante en puissance de calcul et en espace de stockage a conduit à la création d'immenses centres de données. Cependant, leur gestion et leur exploitation consomment d'importantes quantités d'énergie, entraînant non seulement des coûts d'exploitation souvent supérieurs aux investissements initiaux, mais aussi une augmentation significative des émissions de gaz à effet de serre (GES), notamment de dioxyde de carbone.

Les centres de données consomment généralement de grandes quantités d'énergie pour deux raisons. D'une part, les serveurs de stockage ne sont quasiment jamais arrêtés pour éviter de perdre les données confiées au centre de données ou au client cloud. Ils sont ouverts presque 24 heures sur 24, toute l'année. En revanche, ces ordinateurs puissants qui hébergent des machines virtuelle consomment énormément d'énergie pour échapper à l'énorme chaleur générée par leur utilisation, leurs performances sont assurées grâce à une climatisation énergivore.[11]

2.4.1 Répartition de consommation énergétique d'un centre de données

La figure 2.1 montre de manière très éclairante que la source d'énergie la plus consommée par le centre de données est le processeur qui, à lui seul, représente 42 % de la consommation d'énergie. Le second poste de consommation, tout aussi essentiel pour le maintien des performances des équipements est le système de refroidissement, qui consomme 15,4 %. Les disques de stockage et la mémoire vivent respectivement 14,3% et 11,7 % de la consommation. Certaines autres composantes telles que les coûts indirects (7,7 %), les réseaux (4,9 %) et les appareils divers (4 %) impactent aussi la consommation totale. Ce découpage met en avant la nécessité d'optimiser la consommation énergétique des centres de données en utilisant des démarches visant à réduire la consommation d'énergie, réduire les pertes et améliorer les performances de l'infrastructure.[22]

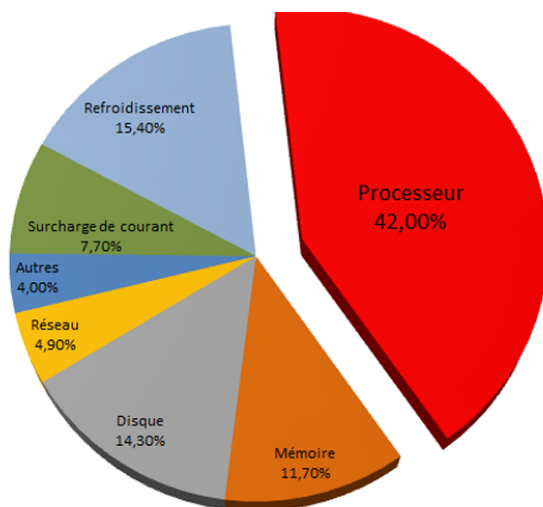


FIGURE 2.1 – Répartition de la consommation énergétique d'un centre de données. [22]

2.5 Techniques de réduction de la consommation d'énergie

La diminution de la consommation d'énergie dans le cloud computing s'appuie sur plusieurs stratégies qui s'efforcent d'augmenter l'usage des ressources tout en préservant une qualité de service adéquate. Ces stratégies sont subdivisées en techniques dynamiques (47 %), techniques statiques (40 %) et autres stratégies (13 %).

Dans ce qui suit, nous allons nous intéresser plus précisément aux techniques statiques et dynamiques qui permettent d'améliorer l'efficacité énergétique des centres de données.

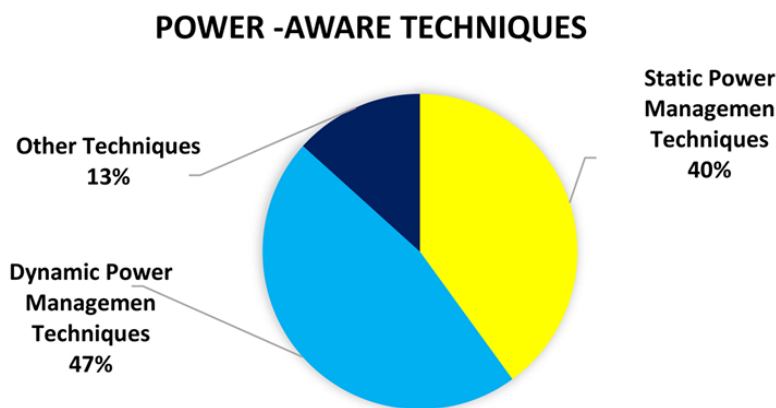


FIGURE 2.2 – Répartition des techniques de gestion de l'énergie.[4]

2.5.1 Gestion statique de l'énergie

Aussi appelée Static Power Management, elle a pour vocation d'optimiser énergétiquement les systèmes en phase de conception. Il s'agit donc de réduire la consommation en intégrant des composants à faible consommation d'énergie ainsi qu'en affinant l'architecture logicielle et matérielle de l'ensemble du système. Cette gestion se pratique donc à deux niveaux : le processeur d'une part, le système d'autre part.[27]

2.5.1.1 Utilisation de matériel informatique écoénergétique

L'une des premières approches d'optimisation énergétique consiste à utiliser du matériel informatique à faible consommation énergétique. Des écolabels tels que US Energy Star, référence nord-américaine de la consommation énergétique des périphériques électroniques, ou la TCO Certification en Europe, qui permet de valoriser du matériel informatique peu énergivore, affirment cette démarche.

Il est à noter que le progrès du matériel contribue à une réduction de la consommation, comme en ce qui concerne le cas des SSD, qui consomment beaucoup moins que les disques durs mécaniques HDD, mais dont les performances sont aussi en grande partie supérieures. En effet, la consommation plus faible de certains composants contribue à la réduction de la consommation électrique globale des systèmes d'information, notamment dans les centres de traitement de données et dans les infrastructures cloud.[3]

2.5.1.2 Optimisation énergétique du CPU et des systèmes

Dans un premier temps, des optimisations au niveau du processeur semblent fondamentales, deux voies principales d'optimisation permettent de réduire la consommation d'énergie du CPU. La première regroupe différentes stratégies d'optimisation au niveau des registres, et devrait contribuer à réduire la consommation d'énergie sur le transfert de registre (RTL) en limitant au maximum les opérations les plus coûteuses en énergie. La seconde type d'approche consiste en des optimisations

dans le cadre de la conception des jeux d'instructions (ISA – Instruction Set Architecture) grâce à certains types d'architectures adaptées, telles que les architectures à largeur de bits restreinte, qui permettent de limiter la consommation énergétique de certaines instructions.

En parallèle de ces améliorations, des stratégies au niveau du système contribuent également à réduire la consommation énergétique. Celles-ci visent à réduire significativement la désynchronisation entre les différents dispositifs (pour qu'ils soient mieux compatibles) et à améliorer la gestion des tâches (des gains sur la charge de travail et la gestion des ressources).

Les choix du système d'exploitation et les choix des logiciels constituent de manière utile une dernière voie pour l'optimisation de la consommation énergétique.[27]

2.5.1.3 C-states

Les C-states, également appelés C-modes, constituent les modes de gestion de l'énergie permettant de mettre le processeur, lorsqu'il est inactif, dans un état de mise en veille ou de repos partiel voire complet. Ces états réduisent progressivement l'alimentation et l'horloge du processeur, offrant une relation entre économie d'énergie et temps de réactivation.

C0 (Operating State) : Etat dans lequel le processeur est actif, consommant de l'énergie au maximum.

C1 (Halt) : Désactivation de l'horloge principale via un logiciel, réduisant ainsi la consommation énergétique tout en permettant un redémarrage rapide.

C2 (Stop Grant and Clock) : Coupure des horloges internes et externes par un mécanisme matériel. Dans ce cas, cette économie d'énergie est plus intéressante, relativement au mode précédent, au prix d'un retour à un état d'opération plus long.

C3 (Sleep) : Stoppe toutes les horloges internes et externes du processeur, abaissant encore plus sa consommation d'énergie.

C4 (Deeper Sleep) : Stoppe toutes les horloges internes du processeur et réduit la tension de l'alimentation au minimum afin de faire tomber au niveau le plus bas la consommation d'énergie.

C5 (Enhanced Deeper Sleep) : Réduit encore le niveau de la tension du processeur et désactive le cache mémoire.

C6 (Deep Power Down) : L'état le plus avancé d'économie d'énergie, c'est-à-dire que la tension interne du processeur est abaissée au minimum, ce qui occasionne donc la plus forte économie d'énergie, mais demande le temps de réactivation le plus long.[14]

2.5.1.4 Optimisation du refroidissement des centres de données

Dans le secteur du cloud computing, se trouve en position prépondérante le refroidissement des centres de mesures pour réduire la perte thermique et donc optimiser la consommation énergétique. Néanmoins, de nombreuses méthodes techniques sont disponibles pour diminuer la consommation d'énergie liée au refroidissement de l'infrastructure informatique, à savoir le free cooling et le refroidissement liquide.

Le free cooling, direct ou indirect, utilise l'air extérieur ou l'eau ambiante comme fluide caloporteur pour refroidir les serveurs lorsque cela est possible. Dans le cas du free cooling direct, l'air extérieur est directement injecté dans le centre de données s'il respecte les critères de qualité, de circulation et d'hygrométrie. Si l'air extérieur ne répond pas aux critères de qualité énoncés précédemment, il doit être prétraité (filtration, déshumidification), ce qui relève alors du free cooling indirect. Cette approche, bien qu'économe en énergie, rejette l'intégralité de la chaleur émise par le centre de données dans l'atmosphère, ce qui représente à coup sûr une perte de chaleur non valorisée.

Un autre levier d'optimisation présenté consiste à mettre en œuvre le refroidissement liquide permettant d'acheminer directement un fluide caloporteur au contact des composants électroniques de l'infrastructure informatique, ce système de refroidissement est plus performant que le refroidissement par air . Un autre axe d'optimisation apparaît être la valorisation d'énergie thermique dissipée en récupérant la chaleur pour la réutiliser dans d'autres opérations industrielles ou pour le chauffage urbain. L'idée est de récupérer cette énergie perdue et de la valoriser comme une ressource prioritairement exploitable, dans une optique de soutenabilité et de réduction de la consommation énergétique du cloud.[21]

2.5.1.5 Consolidation statique de machines virtuelles

Dans la consolidation statiques, les affectations des machines virtuelles aux machines physique ne sont pas modifiés pendant une longue période et aucune migration n'est effectuée avec des changements de charge de travail au cours de cette période .

Cette approche présente l'avantage de permettre le traitement par lots et les applications à demande constante. Un inconvénient de la consolidation statique est le surprovisionnement des ressources. Dans la consolidation statique, les ressources sont allouées de manière à satisfaire les demandes de charge de pointe et par conséquent, elles sont le plus souvent gaspillées la plupart du temps lorsque la machine virtuelle ne fonctionne pas à sa charge de pointe comme dans le cas des centres de données traditionnels.[30]

2.5.2 Gestion dynamique de l'énergie

Aussi connue sous le terme de Dynamic Energy Management, la gestion dynamique permet d'adapter la consommation d'énergie en fonction d'une variation des niveaux de travail à effectuer au sein d'un centre de données. C'est le contraire d'une gestion statique qui s'appuie sur des dispositifs figés sélectionnés lors de la conception. Ce système de gestion de l'énergie présente la possibilité d'évoluer et d'ajuster les ressources selon les besoins du système avec une mise à jour dynamique. Cela signifie par la mise en place d'un suivi permanent de l'état du système et d'un ajustement dynamique du comportement. Cependant, cela nécessite de comprendre le fonctionnement interne du composant mais aussi au-delà comment cet fonctionnement interne interagit avec d'autres composants pour parvenir à un équilibre entre performance et consommation d'énergie optimal.[29]

2.5.2.1 Mise à l'échelle dynamique de la fréquence et de la tension (DVFS)

La mise à l'échelle dynamique de la fréquence et de la tension est une technologie qui ajuste dynamiquement la tension et la fréquence du processeur au moment de l'exécution. En appliquant la technologie DVFS, la tension et la fréquence du système peuvent être ajustées pour s'adapter à différentes tensions de fonctionnement en fonction des spécifications de la conception originale du processeur sans qu'il soit nécessaire de redémarrer l'alimentation. Lorsque le processeur fonctionne à une tension inférieure, il peut effectivement économiser la consommation d'énergie.

La consommation d'énergie dynamique du processeur est modélisée par l'équation suivante :

$$P = V^2 * F * C \quad (6)$$

Où V est la tension, F est la fréquence et C est la charge capacitive du système.

DVFS est une technologie d'économie d'énergie qui consiste à ajuster dynamiquement la tension et la fréquence d'alimentation du processeur. La réduction de la fréquence permet également de diminuer la tension, ce qui entraîne une baisse de la consommation d'énergie, au prix d'une diminution des performances et d'une augmentation des temps d'exécution. De plus, les conditions de surcharge liées à la régulation de la tension doivent être prises en compte.

L'objectif principal du DVFS est de réduire la consommation énergétique en réduisant la vitesse d'exécution des tâches, en fonction de la charge du système. Cette technique est couramment utilisée dans les systèmes embarqués et temps réel, où l'optimisation de l'énergie est cruciale.

De l'équation (6), on peut voir que seule la réduction de la fréquence ne peut pas économiser efficacement l'énergie. En effet, en abaissant la fréquence et en réduisant également la tension, la consommation d'énergie peut être économisée effectivement.

Bien que cette technique n'affecte pas directement la consommation énergétique de la mémoire, du bus, des entrées/sorties et des disques, elle reste une solution efficace pour réduire l'énergie consommée par les processeurs.[28]

2.5.2.2 Ordonnancement des tâches

L'ordonnancement des tâches consiste à organiser dans le temps les tâches provenant des utilisateurs, en tenant compte des contraintes temporelles (contraintes de délai) et des contraintes portant sur l'utilisation et la disponibilité des ressources.

Avant d'aborder le problème d'ordonnancement, il est important de définir les concepts de base du processus d'ordonnancement, tels que les tâches, les ressources. Une tâche est une entité élémentaire localisée dans le temps, par une date de début, une date de fin et une durée préalablement définie. Elle est constituée d'un ensemble d'opérations qui requiert certaines ressources (les machines virtuelles, le CPU et les disques). Ces ressources sont des moyens techniques destinés à être utilisés pour la réalisation d'une tâche et disponibles en quantité limitée. Une ressource est renouvelable si après avoir été allouée à une ou à plusieurs tâches, elle est à nouveau disponible.

Ce processus d'ordonnancement se compose des étapes suivantes :

Phase de priorisation des tâches : Cette phase établit l'ordre des tâches de départ selon leurs priorités. De façon plus précise, les tâches de départ sont celles envoyées par l'utilisateur du cloud et qui arrivent au système d'ordonnancement comme étant le point d'entrée du cloud.

Phase d'allocation des ressources : Cette phase réserve ou alloue un ensemble de ressources nécessaires pour l'ordonnancement des tâches. À cette phase, le contrôleur calcule le nombre de machines virtuelles hébergées dans un ou plusieurs serveurs.

Phase d'ordonnancement : Cette phase sélectionne les ressources (machines virtuelles, CPU, disque, mémoire) parmi celles précédemment allouées. Cette phase fait l'ordonnancement de chaque tâche selon un algorithme qui définit la manière avec laquelle les tâches sont affectées aux machines virtuelles à travers les commutateurs du centre de données.

La manière d'affecter les tâches aux machines virtuelles a un impact sur la consommation d'énergie. Autrement dit, le processus d'ordonnancement joue un rôle important dans la réduction d'énergie consommée. En effet, la consommation d'énergie dépend de la quantité de ressources allouées pendant la phase d'allocation des ressources du processus d'ordonnancement. [8]

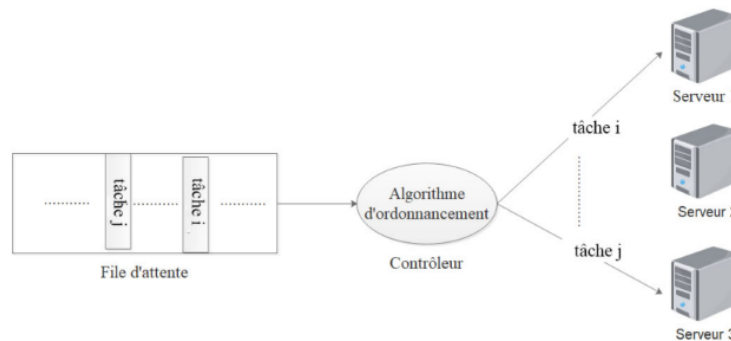


FIGURE 2.3 – Ordonnancement des tâches. [8]

2.5.2.3 Consolidation dynamique de machines virtuelles

Au cœur des centres de données, la virtualisation est devenue l'un des leviers centraux de gestion de la consommation d'énergie. La consolidation de machines virtuelles est l'une des plus répandues. En regroupant les VMs sur un nombre réduit de serveurs, il est alors possible d'éteindre ceux qui ne sont pas utiles, économisant ainsi de l'énergie pour un bon niveau de service. Ce déplacement dynamique des VMs d'un serveur à un autre se fait alors de manière transparente pour les services en cours d'exécution.

De nombreuses approches ont été proposées pour améliorer cette consolidation. L'algorithme MBFD (Modified Best Fit Decreasing) [34], par exemple, cherche à minimiser la consommation d'énergie en attribuant les VMs aux serveurs qui génèrent le plus faible surcoût énergétique, Eco-Cloud [34] utilise un calcul probabiliste qui réduit potentiellement le nombre de migrations de VM

à même de dégrader les performances. D'autres méthodes comme GRANITE [34] peuvent prendre en compte d'autres paramètres, celui de la consommation d'énergie induite par le refroidissement des serveurs, et il existe des approches basées sur l'intelligence artificielle, ainsi LOAD [34] utilise le machine-learning pour anticiper les moments où un serveur va devenir surchargé et adapte de façon dynamique le placement des VMs. Une autre technique, inspirée du comportement des fourmis, est utilisée dans ACS (Ant Colony System) [34], qui permet de trouver une organisation plus efficace des VMs afin de réduire à la fois la consommation d'énergie et le nombre de migrations. Cependant, bien que la consolidation des machines virtuelles contribue à une réduction de la consommation d'énergie, celle-ci doit être menée avec précaution. En effet, un trop grand nombre de migrations risque de pénaliser les applications et au surplus d'engorger le réseau. Le véritable enjeu reste bien de trouver le juste équilibre entre performance d'une part et économie d'énergie d'autre part, ce qui peut se traduire dans le choix de la stratégie la plus pertinente pour chaque centre de données.[34]

2.5.2.4 Power Capping

Le power capping désigne une pratique qui vise à contraindre le niveau de consommation d'énergie au niveau des serveurs d'un centre de données selon un maximum à ne pas dépasser, établissant une limite qui pourrait être inférieure à celle de la consommation maximale des serveurs eux-mêmes, afin de prévenir une sollicitation importante et surtout inefficace du réseau électrique, assurant une gestion cohérente de la consommation d'énergie en fonction de la charge de travail. Dans ce cadre, plusieurs mécanismes de power capping ont été étudiés :

-DPC (Decentralized Power Capping) [17] : Une approche ciblée et décentralisée est utilisée pour réduire la latence et optimiser la consommation énergétique de chaque serveur.

-Dynamo [17], qui est un système développé pour les centres de données de Facebook, repose sur une hiérarchisation de l'énergie et d'un contrôleur de niveau inférieur pour ajuster les consommations des serveurs en fonction du degré de priorité des charges de travail, en mettant en œuvre une allocation heuristique « high-bucket-first » réduisant les impacts négatifs des limitations d'énergie sur les performances des serveurs.

Des chercheurs s'intéressent à plusieurs méthodes de contrôle de la consommation énergétique maximale, et notamment à travers la conception de contrôleurs de rétroaction permettant d'observer périodiquement la consommation énergétique globale et de changer l'état de puissance des serveurs afin de respecter la limite énergétique défini. D'autres travaux portent sur l'allocation optimale de l'énergie au sein des serveurs, en prenant en compte plusieurs paramètres, parmi lesquels la relation puissance-fréquence, le taux d'arrivée des tâches et les configurations de fréquence minimales et maximales des serveurs.

Enfin, la gestion dynamique des limitations énergétiques peut être également mise en œuvre par certains fournisseurs de services de centres de données en participant à un marché ouvert de l'électricité, amenant ainsi à une réduction des coûts et à une baisse de la consommation énergétique.[17]

2.5.2.5 Approvisionnement économe en énergie

Approvisionnement économe en énergie (Energy-aware Provisioning), consiste à allouer les ressources virtuelles (VMs ou conteneurs), intelligemment en fonction des besoins réels des applications afin de minimiser la consommation d'énergie tout en satisfaisant les accords de niveau de service (SLA), ce qui évite la surprovisionnement qui peut entraîner une consommation énergétique accrue et des coûts financiers pour les utilisateurs, ou la sous-provisionnement qui peut potentiellement enfreindre les SLA.

De nombreuses études ont proposé, pour cette allocation, des techniques d'optimisation où la consommation d'énergie est prise en compte. On peut citer par exemple le système EnReal[17] qui procède à un déploiement dynamique de machines virtuelles pour l'optimisation énergétique de l'ensemble. D'autres travaux ont proposé une réponse de type algorithme itératif ou heuristique pour mieux adapter les ressources allouées aux charges de travail.

Certains chercheurs ont également exploré des méthodes basées sur les données pour l'approvisionnement en ressources, en utilisant notamment des techniques de clustering pour regrouper les charges de travail similaires, et des filtres stochastique pour estimer le niveau de charge de travail de chaque catégorie afin d'allouer les ressources de manière efficace sur le plan énergétique. Récemment, Microsoft a proposé le contrôle des ressources (RC) [17] dans le cadre duquel des modèles d'apprentissage automatique ont été entraînés pour fournir des prévisions telles que la durée de vie des machines virtuelles, l'utilisation du processeur et le déploiement maximal des machines virtuelles pour mieux gérer la consommation d'énergie.[17]

2.6 Quelques travaux de recherche sur l'optimisation énergétique des centres de données

Dans cette section, nous analysons plusieurs travaux scientifiques qui traitent des techniques d'optimisation énergétique utilisées dans le cloud computing. L'objectif est de comparer ces travaux, d'identifier les avantages et les inconvénients de chaque technique, ainsi que les métriques d'évaluation utilisées.

2.6.1 Simulateur de mise à l'échelle dynamique de la tension et de la fréquence pour la gestion énergétique des flux de travail réels dans le cloud computing vert

Le travail [9] présente une méthode d'optimisation pour le cloud computing, qui consiste à inclure la technique du DVFS dans les simulations à l'aide du simulateur WorkflowSim. Pour la réalisation de cette démarche, les auteurs ont retenu cinq types d'agents DVFS, c'est-à-dire les dispositifs et stratégies qui mettent en œuvre l'adaptation dynamique de la fréquence et de la tension des processeurs pour la généralité de leur tâche sur la base des utilisateurs à savoir Performance, qui

choisit de maintenir à sa fréquence maximale le processeur pour le réaliser le plus vite possible, Powersave, qui choisit fidèlement de fixer à la fréquence minimale le processeur pour le réduire en même temps de façon à le travailler à basse consommation, Userspace, qui offre à l'utilisateur la possibilité et le choix d'opter manuellement pour une fréquence, OnDemand, qui s'avère capable d'adapter la fréquence de façon dynamique selon l'utilisation effectivement faite du CPU à un instant T et Conservative, qui se peut rendre progressivement de la fréquence de façon à éviter le risque de variation brutale de la fréquence de la machine à un autre moment déterminé.

Des expérimentations ont été réalisées sur plusieurs types de workflows scientifiques authentiques, issus respectivement des projets Montage, Inspiral et Sipht. Un workflow, en effet est une séquence ordonnée de tâches / d'opérations interdépendantes permettant d'accomplir un traitement ou une analyse complexe. Ces workflows contiennent un nombre variable de tâches. Les tests ont été réalisés dans une infrastructure de simulation basée sur des workflows, afin d'évaluer les agents DVFS. Ces derniers ont été validés grâce à une comparaison des courbes de consommation énergétique et des temps d'exécution réels, sous différents agents DVFS. L'évaluation inclut également la méthode d'ordonnancement statique classique (ST-SCH), ainsi que le CAC-SCH, qui combine le DVFS intra-host et inter-host. Cette approche permet une distribution intelligente de la charge afin de minimiser la consommation énergétique globale.

Les résultats des simulations offrent donc bien d'assez nombreuses tendances. En premier lieu, l'emploi de l'ordonnancement CAC-SCH couplé au agent OnDemand a permis de diminuer la consommation énergétique par rapport à d'autres méthodes comme celles basées sur l'ordonnancement statique. En outre, l'algorithme OnDemand a abaissé le temps d'exécution moyen des tâches permettant de garantir d'une part une optimisation conjointe de l'énergie et de la performance, d'autre part a rendu la prévision de la consommation énergétique plus fiable par le modèle énergétique intégré, améliorant ainsi la précision par rapport aux modèles usuels, facilitant aux responsables des centres de données d'adopter des stratégies d'optimisation plus efficaces.[9]

2.6.2 Consolidation auto-adaptative des machines virtuelles pour l'efficacité énergétique dans le cloud

L'article [20] propose l'algorithme SAVE, ce dernier représente une approche auto-adaptative et décentralisée tant au niveau du choix de l'hébergement que de la migration ou de l'arrêt de VM. En effet, alors que les solutions des phases centrales s'appuient sur des logiques déterministes, SAVE s'inspire des modèles probabilistes prenant en compte la dynamique des serveurs qui évaluent précisément l'état de leurs ressources, telle que l'utilisation du CPU, et prennent des choix basés sur une probabilité calculée afin de ne pas migrer de façon systématique des VMs, dont les migrations sont souvent coûteuses, que ce soit en bande passante ou en performances. L'architecture de fonctionnement de SAVE repose sur plusieurs étapes. Tout d'abord, chaque machine physique surveille l'état d'utilisation de ses ressources, puis il doit opter soit pour la possibilité d'héberger plus de VMs sur son propre serveur, soit pour la migration d'un certain nombre d'entre elles, voire même

pour son arrêt afin d'économiser de l'énergie. Puis, une fonction de coût probabiliste est utilisée pour évaluer les migrations en prenant en compte leur coût en énergie et le résultat attendu après leur réalisation. Enfin, seules les migrations apportant une réelle amélioration énergétique sont réalisées, les autres étant évitées pour limiter la surcharge du réseau. L'article a comparé SAVE avec deux solutions populaires : VMware Distributed Resource Scheduler (DRS), une solution commerciale pour gérer dynamiquement l'allocation des VMs et EcoCloud, une approche académique elle aussi visant à réduire la consommation d'énergie en consolidant les VMs. Les résultats expérimentaux montrent que SAVE est plus efficace que ces deux solutions car elle permet de réduire la consommation d'énergie de 30% par rapport à VMware DRS et de gagner entre 10 et 20% par rapport à EcoCloud tout en limitant les migrations inutiles et obtenant une meilleure répartition des charges.[20]

2.6.3 Ordonnement des tâches bi-objectifs axé sur la performance et la consommation d'énergie pour les centres de données cloud

Le travail [23] présente une méthode d'optimisation de l'ordonnement des tâches en centre de données, visant à minimiser la consommation énergétique tout en assurant de bonnes performances. L'objectif est de trouver un équilibre optimal entre performance et consommation d'énergie à l'aide d'un algorithme évolutif basé sur les algorithmes génétiques. Celui-ci optimise l'affectation des charges de travail aux machines virtuelles (VMs) afin d'éviter la sous-utilisation des serveurs et de limiter la consommation excessive d'énergie. L'algorithme évolutif ici utilisé débute par la génération d'une population initiale de solutions au moyen d'une heuristique nommée Modified Worst Fit Decreasing (MWFD), qui répartit intelligemment les tâches pour maximiser l'utilisation des ressources. Il évolue ensuite de manière itérative, évaluant chaque solution selon une fonction bi-objective, intégrant à la fois la consommation d'énergie et le temps d'exécution des tâches. Des opérations de croisement et de mutation permettent d'améliorer les solutions à chaque génération, ne conservant que les configurations les plus efficaces.

L'un des éléments déterminants de cette approche repose sur l'intégration d'un modèle prédictif énergétique performant (LC-MLR) pour l'anticipation conjointe de la consommation des serveurs en fonction de l'utilisation du CPU, de la mémoire, du disque ou du réseau, contrairement aux approches classiques qui n'intègrent que l'utilisation CPU et mémoire, et fournissent une prédiction énergétique plus précise de 9,8 % au bénéfice d'une allocation plus adéquate des ressources et d'économies d'énergie.

Dans le but d'éprouver la pertinence de leur méthode, les auteurs ont mis en œuvre des expériences fondées sur une simulation sous le simulateur CloudSim d'un centre de données fait de 800 hôtes physiques, constitués de plusieurs catégories typées de serveurs hétérogènes, dédiés à un très grand nombre de machines virtuelles (VMs) ayant des principes de configurations différentes, ayant pour but de soumettre l'algorithme à un environnement réaliste assez diversifié. Les charges de travail tirées de PlanetLab sont constituées d'une taille de tâches étant de 6000 à 12500 millions

d'instructions (MI), avec des exigences en CPU, mémoire, disque et réseau aléatoires. Les résultats obtenus montrent que l'algorithme GA_LC-MLR proposé améliore considérablement la consommation énergétique du centre de données par rapport à un algorithme génétique classique (GA_LM). GA_LC-MLR prédit mieux la consommation énergétique des serveurs que GA_LM, car il prend en compte l'utilisation du CPU, de la mémoire, du disque et du réseau, tandis que GA_LM ne considère que le CPU et la mémoire. De plus, GA_LC-MLR optimise la sélection des serveurs en tenant compte de l'exécution parallèle des tâches, minimisant ainsi l'augmentation d'énergie. De plus, les résultats révèlent une réduction du temps d'exécution moyen des tâches, démontrant que l'algorithme exploite efficacement les ressources sans impacter significativement les performances.[23]

2.6.4 Ordonnancement des tâches écoénergétique basé sur le DVFS dans un centre de données.

Le travail [32] présente une approche d'optimisation énergétique basée sur l'ordonnancement des tâches avec DVFS (Dynamic Voltage and Frequency Scaling) afin de réduire la consommation énergétique des centres de données tout en maintenant la qualité de service (QoS). Il apporte également une définition au Energy Consumption Ratio (ECR), qui représente la consommation d'énergie à différentes fréquences en la normalisant par rapport à la consommation à la fréquence maximale. Cette relation forme une fonction convexe, ce qui permet de déterminer une fréquence optimale minimisant la consommation d'énergie tout en respectant les performances requises. Pour résoudre le problème de l'ordonnancement des tâches, des algorithmes heuristiques sont proposés : Le premier, MECRI (Minimum Energy Consumption Ratio Increment), assigne de façon individuelle chaque tâche en minimisant l'augmentation globale d'ECR à partir d'un premier ajustement (« first-fit ») ou, si nécessaire, d'un passage vers une fréquence plus élevée, le second, PTAB (Power-Aware Task Allocation for Batch Tasks), se préoccupe des lots des tâches en combinant une stratégie Best Fit pour les tâches légères, et Worst Fit Decreasing pour les tâches lourdes, réalisant ainsi une préservation de l'équilibre jusqu'à l'efficacité des serveurs. En outre, il propose un algorithme de migration locale des tâches pour réaffecter les tâches en cours dès lors que se termine l'exécution d'une tâche, ce qui permet de baisser davantage l'ECR par un ajustement dynamique de la fréquence sur le même serveur, ce qui permet de bénéficier du faible coût de migration intra-serveur. Les expérimentations comportent des tests sur une base de donnée de test réel constitué de serveurs Dell R720 (avec processeurs ajustables mesurés via un analyseur de puissance) ainsi que des simulations sur de grands clusters. Les expérimentations, menées sur une base de donnée de test réel et via des simulations, montrent des économies d'énergie dépassant 15 % par rapport aux algorithmes classiques (FFD, WFD, MBFD), tout en respectant la QoS et en minimisant les serveurs actifs.[32]

2.6.5 Technologies de free cooling pour les centres de données : mécanisme d'économie d'énergie et applications

Le travail [33] explore les technologies de refroidissement naturel (free cooling) pour réduire la consommation énergétique des centres de données, plusieurs approches de refroidissement ont été étudiées. Parmi elles, le refroidissement direct par air frais consiste à utiliser de l'air extérieur lorsque la température est basse, bien que cela nécessite des dispositifs de purification pour garantir une bonne qualité de l'air intérieur. Ensuite, l'approche de l'échangeur thermique à roue rotative qui évite le mélange d'air en séparant les flux d'air chaud et froid, mais offre un potentiel de refroidissement naturel moindre et occupe plus d'espace. De plus, les racks équipés d'échangeurs thermiques à caloducs intégrant des tubes à transfert de chaleur pour améliorer l'efficacité (3 à 5 fois supérieure aux échangeurs traditionnels). Enfin, le refroidissement par eau repose sur un échangeur thermique couplé à une tour de refroidissement et fonctionne selon trois modes : free cooling en hiver où l'échangeur thermique suffit, hybride au printemps et en automne où il est combiné à un refroidissement actif et électrique en été lorsque seules les unités de climatisation fonctionnent, l'article compare ces différentes technologies en termes de consommation énergétique, coût initial, espace requis et impact sur la qualité de l'air, etc.

En conclusion, le free cooling constitue une solution prometteuse pour réduire la consommation énergétique des centres de données, mais le choix de la technologie doit tenir compte des conditions climatiques locales, des exigences de qualité de l'air et des contraintes budgétaires, tout en combinant plusieurs techniques pour améliorer les performances et minimiser les coûts opérationnels.[33]

2.6.6 Comparaison des travaux de recherche de réduction de la consommation énergétique dans le cloud computing

Ce tableau a pour objet de faire une synthèse des travaux importants sur la réduction de l'énergie en cloud computing, en proposant une analyse comparative des techniques, de leurs avantages et de leurs inconvénients, ainsi que les métriques d'évaluation utilisées.

Réf	Technique	Méthodologie	Avantages	Inconvénients	Métriques d'évaluation
Trav 1 (2017) [9]	DVFS appliqué aux workflows	<ul style="list-style-type: none"> -Extension de WorkflowSim pour intégrer la gestion énergétique via DVFS. -Ajustement de la fréquence en fonction de la charge via divers agents(Performance, Powersave, etc.). -Évaluation expérimentale sur 12 workflows avec différentes tailles (de 25 à 1000 tâches). 	<ul style="list-style-type: none"> -Simulateur réaliste basé sur des workflows scientifiques réels (Montage, Inspiral, Sipt). -Combinaison de stratégies intra-hôtes (DVFS) et inter-hôtes. -Modèles réalistes de consommation d'énergie. -Intégration des coûts de calcul, reconfiguration et réseau. 	<ul style="list-style-type: none"> -Modèle complexe à implémenter. -Performance dépendante de la configuration des agents DVFS. 	<ul style="list-style-type: none"> -Consommation d'énergie. -Temps d'exécution. -puissance total consommée. -Utilisation CPU. -Fréquence DVFS(valeur de fréquence utilisée dynamiquement pour chaque tâche).
Trav 2 (2016) [20]	Consolidation auto-adaptative des VMs (SAVE)	<ul style="list-style-type: none"> -Développement de l'algorithme SAVE pour l'allocation et la migration des VMs. -Évaluation des performances via simulations sous CloudSim avec différentes configurations de machines physiques et VMs. -Comparaison avec EcoCloud et VMWare DRS et SAVE. 	<ul style="list-style-type: none"> -Validation en simulation (CloudSim) et en environnement réel (VMWare), ce qui augmente la crédibilité des résultats. -Réduction significative de la consommation d'énergie. -Simplicité d'implémentation. 	<ul style="list-style-type: none"> -L'approche SAVE n'a été implémentée que partiellement en réel (comparaison avec DRS uniquement, pas EcoCloud). -Nombre élevé de migrations pouvant engendrer des surcoûts. 	<ul style="list-style-type: none"> -Consommation d'énergie. -Nombre de migration. -Utilisation CPU.

<p>Trav 3 (2022) [23]</p>	<p>Planification bi-objectif via algorithme génétique et modèle LC-MLR</p>	<p>-Intégration d'un algorithme évolutif basé sur la métaheuristique génétique pour l'optimisation des tâches. -Génération de la population initiale à l'aide de la stratégie Modified Worst Fit Decreasing (MWFD). -Comparaison avec un algorithme génétique classique utilisant un modèle de puissance plus simple (CPU et mémoire). -Simulation avec CloudSim.</p>	<p>-Optimisation multi-objectifs (temps d'exécution et consommation d'énergie). -S'appuie sur des charges de travail réelles issues du projet PlanetLab (CoMon). -Prise en compte détaillée des ressources (CPU, RAM, disque, réseau). -Réduction énergétique de 3,8% par rapport à GA_LM.</p>	<p>-Complexité de l'algorithme génétique. -Temps de calcul potentiellement long à cause des itérations génétiques.</p>	<p>-Consommation d'énergie. -Temps d'exécution.</p>
<p>Trav 4 (2017) [32]</p>	<p>Planification de tâches avec DVFS</p>	<p>-Définition du ratio de consommation énergétique (ECR). -Algorithme de migration locale des tâches pour optimiser l'énergie. -Algorithmes d'allocation de tâches (MECRI pour les tâches individuelles, PTAB pour les lots de tâches). -Le simulateur est un prototype réel en C++ sous Linux, complété par des tests simulés.</p>	<p>-Réalisation d'expériences en environnement réel et simulation pour comparer la consommation énergétique des algorithmes proposés à d'autres méthodes . -Consommation énergétique optimisée.</p>	<p>-Difficulté à prédire précisément l'énergie consommée sans connaître le temps d'exécution exact. -Complexité algorithmique élevée.</p>	<p>-Consommation d'énergie. -Taux d'acceptation des tâches (nombre de tâches traitées sans rejet). -Nombre de serveurs actifs.</p>

<p>Trav 5 (2017) [33]</p>	<p>Systèmes de refroidissement free cooling</p>	<p>-Sélection de quatre technologies typiques de “free cooling” pour une étude comparative. -Évaluation des caractéristiques techniques de chaque solution (consommation d'énergie , qualité de l'air, etc.). -Aucun simulateur n'est utilisé,l'analyse est basée sur des cas réels expérimentaux.</p>	<p>-Utilisation de sources naturelles (air extérieur, eau) pour diminuer la dépendance aux refroidisseurs électriques. -Réduction significative de la consommation énergétique. -Présentation comparative synthétique des technologies sous forme de tableau récapitulatif.</p>	<p>-Espace requis important pour les échangeurs de chaleur ou les racks à caloduc. -Impact environnemental indirect non analysé (comme l'empreinte carbone). -Nécessité de dispositifs complémentaires (filtration, humidification).</p>	<p>-Consommation d'énergie. -Qualité de l'air. -Coûts. -Investissement primaire. -Espace requis. -Influence de l'humidité intérieure.</p>
--	---	---	---	--	---

TABLE 2.1: Comparaison des travaux de recherche de réduction de la consommation énergétique dans le cloud computing.

2.7 Conclusion

Au fur et à mesure que le cloud computing connaît une forte croissance, la consommation d'énergie, notamment dans les centres de données, est en forte hausse. Ce chapitre a présenté les différentes techniques de réduction de cette consommation énergétique, à travers des présentations des approches statiques et dynamiques, des indicateurs d'efficacité énergétique. Nous avons également exploré les méthodes de gestion de l'énergie comme le DVFS, utilisation de matériel informatique écoénergétique, la consolidation de machines virtuelles et le power capping, et des solutions de système de refroidissement écoénergétiques, etc. En plus, différents travaux de recherche sur la consommation d'énergie dans le cloud computing ont été analysés.

Le chapitre suivant sera, quant à lui, consacré à l'application de certaines de ces techniques à travers une simulation portant sur l'évaluation de leurs impacts sur la consommation d'énergie et la QoS.

CHAPITRE 3

IMPLÉMENTATION ET SIMULATIONS

3.1 Introduction

Ce chapitre expose notre travail pratique sur la réduction de la consommation d'énergie dans le cloud, en mettant en œuvre et en simulant plusieurs approches telles que le DVFS qui permet d'ajuster dynamiquement la fréquence et la tension, la consolidation dynamique des machines virtuelles, ainsi qu'une approche qui combine les deux procédés. La première partie du chapitre sera consacrée à la présentation de l'environnement de travail utilisé, ainsi qu'à la description du langage de programmation Python, les bibliothèques que nous utilisons pour le développement, les paramètres techniques de notre simulateur.

La deuxième partie du chapitre consacrée au détail des trois mécanismes d'optimisation énergétique mis en œuvre dans notre simulation, y compris le DVFS, la consolidation des VMs, et la méthode combinée. Leurs pseudocodes et leurs principes de fonctionnement seront introduits dans cette partie. Enfin, nous détaillerons les scénarios expérimentaux appliqués, et nous donnerons et analyserons les résultats obtenus à travers plusieurs scénarios, tout en comparant chaque technique selon la quantité d'énergie consommée, le respect des délais et le temps d'exécution.

3.2 Outils et l'environnement de développement

Le travail proposé dans ce mémoire a été implémenté et testé dans un environnement possédant les caractéristiques suivantes :

- Une machine avec un processeur Intel (R) Core (TM) i5-1145G7 CPU@ 2.60GHz, une vitesse de 2.61 GHz et une capacité mémoire de 16,0 GO.
- Le simulateur Simpy est sous Windows 11 de 64 bits.
- Le simulateur Simpy version 4.1.1.
- Le langage de programmation Python.
- L'IDE Visual Studio Code .

3.2.1 Langage de programmation Python

Python est un langage de programmation open-source puissant et facile à apprendre. Il possède des structures de données de haut niveau efficaces et une approche simple mais efficace de la programmation orientée objet. La syntaxe élégante et le typage dynamique de Python, ainsi que sa nature interprétée, en font un langage idéal pour l'écriture de scripts et le développement rapide d'applications dans de nombreux domaines et sur la plupart des plates-formes. L'interpréteur Python et la vaste bibliothèque standard sont disponibles gratuitement sous forme de source ou de binaire pour toutes les principales plates-formes.[45]

3.2.2 Visual Studio Code

Visual Studio Code (VSCoDe) est un éditeur de code source et un environnement de développement intégré (IDE) de Microsoft. Il est open-source et cross-platform, c'est-à-dire qu'il fonctionne sur Windows, Linux et Mac. Il a été conçu pour les développeurs web, mais il prend en charge de nombreux autres langages de programmation tels que C++, C, Python, Java, etc. Il offre de nombreuses fonctionnalités comme la coloration syntaxique, l'auto-complétion, la mise en évidence des erreurs, la navigation de code, le débogage, la gestion de versions, l'intégration avec Git, et beaucoup d'autres. Il est également extensible à l'aide d'une grande variété d'extensions développées par la communauté, permettant aux développeurs de personnaliser l'éditeur selon leurs besoins.[37]

3.2.3 Bibliothèques Python utilisées

Random : Random est un module Python regroupant plusieurs fonctions permettant de travailler avec des valeurs aléatoires. La distribution des nombres aléatoires est réalisée par le générateur de nombres pseudo-aléatoires Mersenne Twister, l'un des générateurs les plus testés et utilisés dans le monde informatique.[42]

Tqdm : Tqdm est une bibliothèque Python qui fournit une barre de progression rapide et extensible pour les boucles et les itérables, facilitant la visualisation de l'avancement des traitements dans un programme.[47]

Matplotlib : Matplotlib est une bibliothèque Python open-source. Son objectif initial était de reproduire les fonctionnalités graphiques de MATLAB en Python. Aujourd'hui, elle est devenue la référence pour la création de graphiques, animés et interactifs dans l'écosystème Python.[43]

Enum : Le module enum est un module standard qui permet de créer des énumérations, c'est-à-dire un ensemble de noms symboliques associés à des valeurs constantes. Les énumérations permettent d'écrire du code plus lisible, structuré, en remplaçant des constantes magiques (valeurs brutes) par des noms significatifs.[41]

Time : Le module time est un module standard de Python qui fournit des fonctions pour manipuler le temps et les horloges système. Il permet notamment de mesurer la durée d'exécution d'un programme, de suspendre l'exécution d'un script pendant un temps donné ou de récupérer l'heure actuelle (time()). Ce module est couramment utilisé dans les simulations, les boucles d'attente, les mesures de performances, ou encore les animations temporelles.[46]

Simpy : Simpy est une bibliothèque de simulation de modélisation d'événements discrets dans le langage Python pour simuler des systèmes dynamiques (file d'attente, réseau, infrastructures cloud, etc). En effet, Simpy met en œuvre un temps de simulation virtuel et variable, qui permet également de simuler rapidement des systèmes qui exigent dans la réalité un temps bien plus long pour être simulés. La simulation de systèmes cloud peut être abordée sous plusieurs approches , par exemple, Simpy et CloudSim, le premier présente un certain nombre d'avantages par rapport

au second, tels que la flexibilité de modélisation, la simplicité de syntaxe du langage Python, et la possibilité d'une intégration facile avec des bibliothèques d'analyse et de visualisation de données comme Pandas et Matplotlib, permettant de réaliser des simulations faciles de scénarios complexes sans imposer une structure rigide, facilitant ainsi l'expérimentation de stratégies personnalisées, dont notamment celles sur l'optimisation énergétique.[44]

3.2.4 Paramètres de l'environnement

Plusieurs paramètres ont été définis dans le cadre de l'environnement de simulation au sein de cette étude, incluant des caractéristiques relatives aux machines physiques et virtuelles, aux tâches ainsi que les paramètres de base.

Pour les machines physiques, elles sont caractérisées par la vitesse du processeur (MIPS, Millions d'Instructions Par Seconde), c'est-à-dire le nombre de millions d'instructions qu'il peut exécuter en une seconde. C'est une façon de quantifier la puissance de traitement d'un CPU. Les machines physiques sont aussi définies par leur nombre de cœurs et leur mémoire vive (RAM, en Go).

Pour les machines virtuelles, les mêmes types de paramètres sont pris en compte, mais avec des valeurs réduites.

Les tâches sont définies selon leur longueur (en millions d'instructions, MI) et le nombre de cœurs nécessaires pour l'exécution. L'ensemble des valeurs est choisi aléatoirement dans des intervalles spécifiés.

Paramètres	Valeurs
Vitesse du processeur (MIPS)	10000, 12000, 8000
Nombre de coeurs	8 , 12 , 16
Mémoire vive (Go)	16 , 32

TABLEAU 3.1 – Paramètres des machines physiques

Paramètres	Valeurs
Vitesse du processeur (MIPS)	[500-2000]
Nombre de coeurs	[1-2]
Mémoire vive (Go)	1 , 2 , 4

TABLEAU 3.2 – Paramètres des machines virtuelles

Paramètres	Valeurs
Taille des tâche (MI)	[10000-500000]
Nombre de coeurs	1

TABLEAU 3.3 – Paramètres des tâches

Paramètres	Valeurs
Nombre des centres de données	3
Nombre des machines physiques dans chaque centre de données	10
Nombre total des machines physiques	30
Nombre des VMs	100
Nombre des tâches	500 , 1000 , 1500 , 2000

TABLEAU 3.4 – Paramètres de base

3.3 Calcul de l'énergie consommée

Le calcul de l'énergie repose sur un modèle détaillé prenant en compte plusieurs paramètres. Pour chaque centre de données, l'énergie totale est cumulée par tranches de temps (*time_slice*) à l'aide d'une méthode, qui multiplie la puissance instantanée totale par la durée de la tranche. La puissance d'un hôte est déterminée par une méthode selon son état (50W lorsqu'il est inactif), son taux d'utilisation (rapport entre MIPS utilisés et MIPS totaux), et son niveau de puissance. Chaque hôte peut fonctionner à trois niveaux (HIGH, MEDIUM, LOW), définis par des combinaisons de fréquence (2.5 / 2.0 / 1.5 GHz), de tension (1.2 / 1.0 / 0.8 V), et d'un coefficient de puissance (1.0 / 0.8 / 0.6). Initialement, tous les hôtes sont configurés au niveau de puissance HIGH. Par la suite, les stratégies d'optimisation appliquent des ajustements dynamiques.

Le modèle introduit un facteur $\alpha = 2$ pour simuler la relation entre l'utilisation des ressources et la consommation d'énergie, selon la formule suivante :

$$\text{puissance} = p_{\text{idle}} + (p_{\text{max}} - p_{\text{idle}}) \times (\text{utilisation}^\alpha) + \text{consommation_composants_base} \quad (7)$$

où :

p_{idle} : Puissance consommée lorsque le serveur est au repos.

p_{max} : Puissance consommée lorsque le CPU est utilisé à 100 %.

utilisation : Charge du CPU, une valeur comprise entre 0 et 1.

consommation_composants_base : Puissance consommée par les autres composants (RAM, disque, etc).

Les valeurs de p_{idle} et p_{max} sont ajustés en fonction du facteur de puissance, calculé comme suit :

$$\text{facteur_puissance} = \frac{\text{tension}^2 \times \text{fréquence}}{\text{tension}_{\text{HIGH}}^2 \times \text{fréquence}_{\text{HIGH}}} \times \text{coefficient_puissance} \quad (8)$$

Une consommation fixe de 20W est ajoutée pour les composants de base. La consommation énergétique totale est calculée séparément dans chaque tranche de temps (60 secondes), puis ajoutée à

la consommation énergétique totale du centre de données par la formule suivante :

$$\text{Énergie} = \text{Puissance} \times \text{Temps} \quad (9)$$

Ce modèle permet une estimation fine et réaliste de la consommation énergétique dans un environnement cloud, en prenant en compte les caractéristiques matérielles et le niveau d'activité des hôtes.

3.4 Approches utilisés pour l'optimisation énergétique

Dans le contexte de notre projet de fin d'études portant sur la réduction de la consommation d'énergie dans le cloud computing, différentes approches d'optimisation sont appliquées et testées, à savoir l'ajustement dynamique de la fréquence et de la tension (Dynamic Voltage Frequency Scaling-DVFS), la consolidation des machines virtuelles (VM Consolidation), ainsi qu'une stratégie combinant les deux techniques (Combinée).

3.4.1 Mise à l'échelle dynamique de la tension et de la fréquence (DVFS)

Le DVFS (Dynamic Voltage and Frequency Scaling) consiste à faire varier dynamiquement, en fonction de l'utilisation du processeur, la fréquence d'horloge ainsi que sa tension d'alimentation. Dans la simulation de base que nous avons choisi d'utiliser, il se positionne au niveau des hôtes et s'exprime par trois niveaux de puissance prédéfinis (PowerLevel) HIGH, MEDIUM, LOW, chacun d'eux possède un ensemble de caractéristiques (fréquence, tension, coefficient de puissance) influençant directement la consommation d'énergie .

Un mécanisme DVFS est intégré dans le simulateur afin d'ajuster dynamiquement la consommation d'énergie des hôtes en fonction de leur charge réelle. Au début de la simulation, tous les hôtes démarrent à un niveau HIGH, avec une tension et une fréquence maximales, haute consommation d'énergie assurée. Lors de chaque pas de simulation, si DVFS est activé l'algorithme ajuste le niveau de puissance de chaque hôte actif selon son taux d'utilisation des ressources :

- Lorsque l'utilisation tombe en dessous de 30%, le système passe en mode **LOW** (1.5 GHz, 0.8V, 1.0).
- Lorsque l'utilisation est comprise entre 30% et 70%, il passe en mode **MEDIUM** (2.0 GHz, 1.0V, 0.8).
- Lorsque l'utilisation dépasse 70%, le système reste en mode **HIGH** (2.5 GHz, 1.2V, 0.6).

L'intérêt du DVFS repose alors sur sa capacité à diminuer fortement la puissance lorsque les ressources sont sous-utilisées, grâce à une réduction de la fréquence et de la tension de fonctionnement

du processeur. Dans ce cas, l'utilisation de l'énergie est alors optimisée et la perte d'énergie éventuellement subie est réduite, puisque l'on travaille à un niveau d'exploitation des ressources plus adapté.

3.4.2 Consolidation dynamique des VMs

La consolidation des machines virtuelles est une méthode incontournable à mettre en oeuvre dans les centres de données cloud pour diminuer la consommation d'énergie tout en conservant un niveau de performance suffisant. La consolidation des machines virtuelles peut en effet être vue dans le cadre de notre simulation, comme un moyen d'optimiser l'utilisation des ressources physiques des centres de données en activant le moins d'hôtes possible . Elle est exécutée de façon périodique (toutes les 600 secondes dans cette configuration) et suit une méthodologie bien définie :

- Identification des hôtes sous-utilisés : Le système classe les hôtes actifs par taux d'utilisation croissant afin de sélectionner les hôtes dont le taux d'utilisation est inférieur à 20%. Les hôtes éligibles pour les migrations de leurs VMs sont ceux dont le taux d'utilisation est en dessous du seuil. Ce dernier sert à mettre en évidence les machines où l'on consomme de l'énergie inutilement tout en laissant une marge acceptable afin d'éviter des migrations inutiles.
- Choix des hôtes cibles : Les hôtes dont le taux d'utilisation est inférieur à 80% sont considérés comme éligibles pour être sélectionnés comme candidats potentiels pour les migrations et sont d'abord ordonnés par taux d'utilisation décroissant pour garantir une bonne répartition de charge de travail.
- Migration des VMS : Pour chaque VM positionnée sur l'hôte sous-utilisé, l'algorithme cherche un autre hôte pouvant accueillir cette VM. Il y a trois conditions à vérifier pour qu'un hôte soit capable d'accueillir une VM, à savoir :
 - La mémoire libre de l'hôte doit être suffisante : $host.available_memory \geq vm.memory$
 - Le nombre de cœurs libre de l'hôte doit être suffisant : $host.available_cores \geq vm.cores$
 - Le nombre de MIPS disponible doit être suffisant : $(host.total_mips - host.allocated_mips) \geq vm.mips$). Si une VM trouve un hôte capable de l'accueillir, elle est migrée , sinon elle reste sur son hôte actuel.
- Désactivation des hôtes vides : Si un hôte se retrouve sans aucune VM après migration, son niveau de puissance doit être abaissé (PowerLevel.LOW) pour diminuer sa consommation à 50W.
- Optimisation continue : Ce processus se déclenche périodiquement (toutes les 600 secondes dans la simulation) pour s'adapter de manière continue aux variations de charge de travail dans le centre de données.

3.4.3 Combinaison des techniques DVFS et de consolidation

La méthode combinée représente une stratégie globale d'optimisation énergétique, basée sur la combinaison des deux techniques DVFS et consolidation de machines virtuelles, prenant appui sur les synergies de ces deux techniques complémentaires. En effet, cette méthode permet quand elle est activée au DVFS et à la consolidation de fonctionner de manière synchrone, le DVFS optimise la consommation des serveurs encore actifs en faisant varier en temps réel leurs niveaux de puissance, tandis que la consolidation mène à une réorganisation périodique de l'ensemble des serveurs. Cette synergie permet de créer un cercle vertueux, la consolidation limitant le nombre de serveurs actifs, ce qui rend le fonctionnement du DVFS plus adapté.

3.5 Pseudocodes des mécanismes d'optimisation énergétique

3.5.1 Pseudocode de l'algorithme DVFS

Algorithm 1 : Algorithme de DVFS

Entrées : centres_données et leurs hôtes

Sorties : niveau_puissance_hotes

1. PowerLevel.HIGH \leftarrow (fréquence : 2.5, tension : 1.2, coefficient : 1.0)
2. PowerLevel.MEDIUM \leftarrow (fréquence : 2.0, tension : 1.0, coefficient : 0.8)
3. PowerLevel.LOW \leftarrow (fréquence : 1.5, tension : 0.8, coefficient : 0.6)

4. **Fonction** AppliquerDVFS(centres_données)
5. niveau_puissance_hotes \leftarrow []
6. seuil_bas \leftarrow 0.3
7. seuil_moyen \leftarrow 0.7
8. **Pour chaque** centre_de_données **dans** centres_données **faire**
9. hotes_actifs \leftarrow []
10. **Pour chaque** hote **dans** centre_de_données.hotes **faire**
11. **Si** hote.est_actif = Vrai **alors**
12. Ajouter hote à hotes_actifs
13. **Fin Si**
14. **Fin Pour**
15. **Pour chaque** hote **dans** hotes_actifs **faire**
16. **Si** hote.mips_total > 0 **alors**

```
17.      utilisation ← hote.mips_alloués / hote.mips_total
18.      utilisation ← MAX(0, MIN(1, utilisation))
19.      Sinon
20.      utilisation ← 0
21.      Fin Si
22.      Si utilisation < seuil_bas alors
23.      nouveau_niveau ← PowerLevel.LOW
24.      Sinon si utilisation < seuil_moyen alors
25.      nouveau_niveau ← PowerLevel.MEDIUM
26.      Sinon
27.      nouveau_niveau ← PowerLevel.HIGH
28.      Fin Si
29.      hote.power_level ← nouveau_niveau
30.      valeurs_hote ← (hote.power_level.frequence, hote.power_level.tension, hote.power_level
    .coefficient)
31.      Ajouter valeurs_hote à niveau_puissance_hotes
32.      Fin Pour
33.      Fin Pour
34.      Retourner niveau_puissance_hotes
35. Fin Fonction
36. Début
37.  niveau_puissance_hotes ← AppliquerDVFS(centres_données)
38.  Retourner niveau_puissance_hotes
39. Fin
```

3.5.2 Pseudocode de l'algorithme de consolidation

Algorithm 2 : Algorithme de consolidation

Entrées : centres_données, hotes, temps_actuel, dernière_consolidation, Consolidation_Interval.

Sorties : vms_migrées_total, hotes_vidés.

```
1. Si temps_actuel < dernière_consolidation + Consolidation_Interval alors
2.   Retourner 0
3. Fin Si
4. dernière_consolidation ← temps_actuel
5. Fonction ConsoliderDatacenters(centres_données)
6.  tous_hotes ← []
```

```

7.  Pour chaque centre_de_données dans centres_données faire
8.    Pour chaque hote dans centre_de_données faire
9.      Ajouter hote à tous_hotes
10.   Fin Pour
11.  Fin Pour
12.  hotes_par_utilisation ← TRIER( Pour chaque hote dans tous_hotes si hote.est_actif
ET hote.vms ≠ vide, par hote.get_utilization() croissant)
13.  hotes_cibles ← Pour chaque hote dans tous_hotes si hote.est_actif
ET hote.get_utilization() < 0.8
14.  hotes_cibles ← TRIER(hotes_cibles, par hote.get_utilization() décroissant)
15.  vms_migrées_total ← 0
16.  hotes_vidés ← 0
17.  seuil_sous_utilisation ← 0.2
18.  Pour chaque hote_source dans hotes_par_utilisation faire
19.    Si hote_source.vms = vide OU hote_source.get_utilization() ≥ seuil_sous_utilisation
alors
20.      Continuer
21.    Fin Si
22.    vms_à_migrer ← COPIE(hote_source.vms)
23.    toutes_vms_migrées ← Vrai
24.    Pour chaque vm dans vms_à_migrer faire
25.      vm_migrée ← Faux
26.      Pour chaque hote_destination dans hotes_cibles faire
27.        Si hote_destination = hote_source alors
28.          Continuer
29.        Fin Si
30.        Si (hote_destination.mémoire_disponible ≥ vm.mémoire) ET
(hote_destination.cores_disponibles ≥ vm.cores) ET
(hote_destination.mips_total - hote_destination.mips_alloués ≥ vm.mips) alors
31.          Ajouter vm à hote_destination, mettre à jour ses ressources
32.          Si vm.hote = hote_destination alors
33.            vm_migrée ← Vrai
34.            vms_migrées_total += 1
35.          Sinon
36.            Afficher erreur migration
37.            toutes_vms_migrées ← Faux
38.          Fin Si
39.        Fin Si
40.      Fin Pour

```

```
41.     Si vm_migrée = Faux alors
42.         toutes_vms_migrées ← Faux
43.     Fin Si
44. Fin Pour
45. Si toutes_vms_migrées = Vrai ET hote_source.vms = vide alors
46.     hote_source.est_actif ← Faux
47.     hote_source.set_power_level(PowerLevel.LOW)
48.     hotes_vidés += 1
49.     Si hote_source dans hotes_cibles alors
50.         Supprimer hote_source de hotes_cibles
51.     Fin Si
52. Fin Si
53. Fin Pour
54. Retourner vms_migrées_total, hotes_vidés
55. Fin Fonction
56. Début

57. vms_migrées_total, hotes_vidés ← ConsoliderDatacenters(centres_données)
58. Retourner vms_migrées_total, hotes_vidés
59. Fin
```

3.5.3 Pseudocode de l'algorithme combiné DVFS et consolidation

Algorithm 3 : Méthode combinée DVFS et consolidation

Entrées : centres_données, hotes, temps_actuel, dernière_consolidation, Consolidation_Interval.

Sorties : niveau_puissance_hotes, vms_migrées_total, hotes_vidés.

1. Début

2. vms_migrées_total ← 0

3. hotes_vidés ← 0

4. niveau_puissance_hotes ← AppliquerDVFS(centres_données)

5. Si temps_actuel \geq dernière_consolidation + Consolidation_Interval alors

6. vms_migrées_total, hotes_vidés ← ConsoliderDatacenters(centres_données)

7. dernière_consolidation ← temps_actuel

8. Sinon

9. vms_migrées_total ← 0

10. hotes_vidés ← 0

11. **Fin Si**
12. **Retourner** niveau_puissance_hotes, vms_migrées_total, hotes_vidés
13. **Fin**

3.6 Résultats et analyses

3.6.1 Résultats de simulation pour 500 tâches

Pour ce tableau, la simulation a été effectuée avec 500 tâches. Pour le scénario sans optimisation, la consommation d'énergie est de 5469,44 Wh, pour un temps d'exécution de 3960 secondes, respect de délai de 50,80 %. L'application de la technique DVFS permet de réduire significativement la consommation d'énergie, atteignant 2817,16 Wh, soit presque 49 %, avec un léger bénéfice du respect de délai qui passe à 52,00 %, pour un temps d'exécution légèrement plus court (3840 s). Le scénario de consolidation, quant à lui, entraîne également une baisse de consommation d'énergie (4469,19 Wh) et un gain sur le temps d'exécution (3480 s), mais le respect de délai est faiblement dégradé, avec un taux de 48,40 %. Enfin, le scénario DVFS + consolidation (méthode combinée) permet une consommation modérée (3801,59 Wh), un temps d'exécution plus long (4620 s), et un respect de délais très proche de celui du scénario de base. (50,40 %), tout en conservant, dans chacun des cas, le nombre de tâches exécutées égal à 500/500.

Scenario	Énergie (Wh)	Temps d'exécution (s)	Respect de délai (%)	Tâches terminées
non optimisé	5469.44	3960.0	50.80	500/500
dvfs	2817.16	3840.0	52.00	500/500
consolidation	4469.19	3480.0	48.40	500/500
combinée	3801.59	4620.0	50.40	500/500

TABLEAU 3.5 – Résultats pour 500 tâches.

3.6.2 Résultats de simulation pour 1000 tâches

Ce tableau a pour modèle une charge de travail de 1000 tâches, soit la configuration initiale de l'étude. Le scénario non optimisé fournit un résultat de 9648,29 Wh, avec un temps d'exécution de 7380 secondes, et un respect de délai de 11,70 %. La mise en œuvre de DVFS réduit l'énergie d'environ la moitié à 5051,09 Wh, et réduit également le temps d'exécution à 6540 secondes, le respect de délai connaît une légère diminution, atteignant 10,00 %. En revanche, en cas de consolidation seulement, une légère amélioration de la qualité de service est observée, avec 13,60 % des tâches achevées dans les délais prévus, pour une consommation énergétique de 9079,28 Wh et un

temps d'exécution de 7020 secondes. Le scénario combiné DVFS + consolidation arrive à trouver une sorte d'équilibre : 6737,49 Wh d'énergie consommés (une réduction de 30,16 % par rapport au scénario non optimisé), un temps d'exécution croissant (8040 s) mais un taux de respect de délais légèrement supérieur est observé, atteignant 12,30 %, comparé aux scénarios non optimisé et avec DVFS uniquement.

Scenario	Énergie (Wh)	Temps d'exécution (s)	Respect de délai (%)	Tâches terminées
non optimisé	9648.29	7380.0	11.70	1000/1000
dvfs	5051.09	6540.0	10.00	1000/1000
consolidation	9079.28	7020.0	13.60	1000/1000
combinée	6737.49	8040.0	12.30	1000/1000

TABLEAU 3.6 – Résultats pour 1000 tâches.

3.6.3 Résultats de simulation pour 1500 tâches

Dans ce tableau , la simulation traite 1500 tâches. Le scénario non optimisé consomme bien 14740,17 Wh sur 11460 secondes, avec un respect du délai très faible, 1,93 % alors que toutes les tâches sont effectuées. L'application du DVFS permet de réduire de moitié la consommation d'énergie,atteignant 6972,41 Wh, avec un temps d'exécution réduit à 9420 secondes, avec une légère amélioration du taux de respect des délais, atteignant 2,27 %, par rapport au scénario non optimisé.La consolidation permet une légère amélioration du respect des délais, atteignant 2,33 %, pour une consommation de 11989,37 Wh et un temps d'exécution de 9780 secondes. Enfin, la combinaison des deux techniques conduit à une consommation contenue (7501,56 Wh) et un temps d'exécution réduit (9000 s), mais cela s'accompagne d'une légère dégradation du respect de délais, avec un taux de 1,13 %, le plus faible enregistré, et là encore en ne tenant compte que des tâches effectivement exécutées (1500/1500).

Scenario	Énergie (Wh)	Temps d'exécution (s)	Respect de délai (%)	Tâches terminées
non optimisé	14740.17	11460.0	1.93	1500/1500
dvfs	6972.41	9420.0	2.27	1500/1000
consolidation	11989.37	9780.0	2.33	1500/1500
combinée	7501.56	9000.0	1.13	1500/1500

TABLEAU 3.7 – Résultats pour 1500 tâches.

3.6.4 Résultats de simulation pour 2000 tâches

Le dernier tableau testé avec 2000 tâches présente la consommation d'énergie de 16844,43 Wh en mode non optimisé avec un temps d'exécution de 13260 s et un taux de respect de délai extrêmement bas de 0,15 %, toutes les tâches ayant été exécutées (2000/2000). Les ajouts du DVFS ont permis une abaissement significatif de la consommation à 10463,06 Wh mais avec une légère augmentation du temps d'exécution (13320 s), et cette situation présente également une légère augmentation du taux de respect des délais, atteignant 0,55 %. Avec la stricte consolidation, la consommation d'énergie ne s'élève qu'à 13140,81 Wh avec le temps d'exécution le plus court de tous les scénarios (10680 s) mais le taux de respect de délai s'élève à 0,25 % (une légère amélioration). Enfin, le scénario combiné atteint la meilleure performance énergétique globale avec une seule consommation de 8746,41 Wh, soit presque la moitié de non optimisé et un temps d'exécution modéré (11640 s). Le taux de respect de délais reste limité, mais enregistre une légère amélioration à 0,25 %. et autorise à l'inverse un bon compromis entre des économies globales d'énergie et globales de performances.

Scenario	Énergie (Wh)	Temps d'exécution (s)	Respect de délai(%)	Tâches terminées
non optimisé	16844.43	13260.0	0.15	2000/2000
dvfs	10463.06	13320.0	0.55	2000/2000
consolidation	13140.81	10680.0	0.25	2000/2000
combinée	8746.41	11640.0	0.25	2000/2000

TABLEAU 3.8 – Résultats pour 2000 tâches.

3.6.5 Consommation d'énergie par rapport au nombre de tâches

L'augmentation du nombre de tâches exerce une influence claire sur la consommation énergétique des centres de données simulés. En effet, plus il y a de tâches, plus la demande en ressources sur les hôtes physiques et sur les VMs est importante et donc consomme plus d'énergie.

En général, les résultats montrent effectivement que la consommation d'énergie augmente avec le nombre de tâches, surtout dans le cadre du scénario non optimisé où la consommation dépasse les 16 000 Wh pour 2 000 tâches soumis au système. Avec les techniques d'optimisation DVFS, consolidation et la méthode combinée, nous sommes en mesure de réduire significativement la consommation énergétique dans tous les cas de charge de travail, mais cela ne nous empêche pas d'assister à une consommation qui augmente selon le nombre de tâches lorsque celui-ci devient trop grand. En analysant les résultats pour chaque palier de charge de travail (500, 1000, 1500 et 2000 tâches), on note que la technique DVFS est la plus efficace pour 500, 1000 et 1500 tâches avec un taux de réduction respectivement de 48,49 %, 47,65 % et 52,70 % par rapport au scénario

non optimisé, présentant sur ces scénarios les consommations d'énergie les plus faibles, car elle est en mesure de faire évoluer dynamiquement les tensions et fréquences processeurs en fonction des charges de travail permettant de mieux optimiser la consommation énergétique lorsque les charges de travail sont modérées. En revanche, pour 2000 tâches, c'est la technique combinée (DVFS + consolidation) qui se révèle alors plus efficace (avec un taux de réduction de 48,07 % par rapport au scénario non optimisé). En effet, cette combinaison nous permet d'avoir à la fois une dynamique des ressources et une réduction du nombre de machines actives, ce qui va se révéler particulièrement efficace lorsque les charges deviennent très importantes.

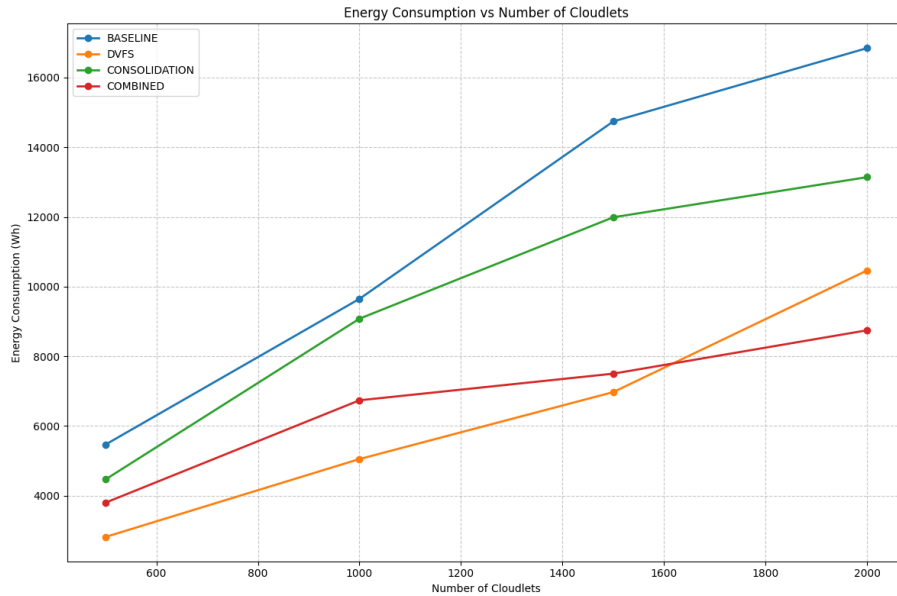


FIGURE 3.1 – Consommation d'énergie par rapport au nombre de tâches.

3.6.6 Temps d'exécution par rapport au nombre des tâches

Le graphique montre l'impact des différentes techniques de réduction de la consommation d'énergie sur le temps d'exécution pour 500, 1000, 1500 et 2000 tâches. Pour 500 tâches, les méthodes DVFS et consolidation permettent un temps d'exécution plus court que le scénario sans optimisation, avec consolidation étant la plus rapide. La méthode combinée est légèrement plus lente à ce stade. Pour 1000 tâches, la méthode combinée reste la plus lente des méthodes, tandis que DVFS et consolidation conservent une bonne performance. À 1500 tâches, DVFS reste plus rapide que consolidation, la méthode combinée présente de meilleurs résultats que le DVFS en ce qui concerne le temps d'exécution. Enfin, à 2000 tâches effectuées, DVFS a un temps d'exécution pratiquement identique à la méthode de base, confirmant que sa stratégie énergétique peut se révéler contre-productive dans le cas d'une forte charge. Consolidation s'avère alors la plus performante, avec le plus court temps d'exécution.

L'analyse de ces résultats permet d'identifier les effets de chaque technique concernant le temps

d'exécution, avec DVFS, le temps d'exécution diminue en charge de travail modérée, ce qui contredit l'idée selon laquelle une baisse de fréquence CPU allonge toujours le temps. Cela s'explique par la nature indépendante des tâches, qui peuvent être traitées efficacement même à fréquence réduite. En revanche, à très forte charge, le temps d'exécution connaît une légère hausse, presque négligeable, traduisant un début d'effet défavorable lié à la baisse de la fréquence CPU.

Pour ce qui est de la consolidation, alors que la technique peut théoriquement générer une surcharge des serveurs actifs, les résultats montrent qu'on obtient très souvent un temps d'exécution plus faible, la nature indépendante des tâches assurant la bonne gestion des VMs sans ralentissement. Pour la méthode combinée, l'exécution est généralement plus longue, en raison d'une application en parallèle de la méthode DVFS ainsi que de la technique de consolidation. Pourtant, grâce à l'indépendance des tâches toujours, l'augmentation du temps d'exécution est parfois compensée, rendant ainsi l'exécution plus harmonieuse à forte charge de travail.

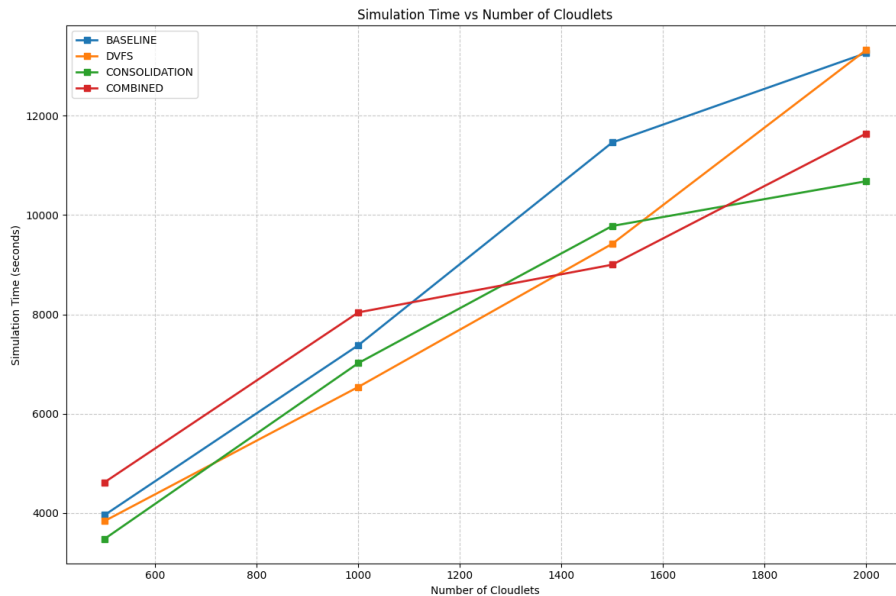


FIGURE 3.2 – Temps d'exécution par rapport au nombre des tâches.

3.6.7 Respect de délai par rapport au nombre de tâches

L'analyse du graphique montre que les techniques de réduction de la consommation d'énergie impactent légèrement, mais de façon différenciée le respect de délais, en fonction de la charge de travail. Pour 500 tâches, et parmi les méthodes que nous appliquons DVFS, consolidation ou leur combinaison, le respect de délais est légèrement affecté, mais le niveau de réussite reste relativement élevé (entre 48 % et 52 %). En revanche, à 1000 tâches, l'application du DVFS provoque une légère baisse du respect des délais, atteignant environ 10 % tandis que consolidation et la méthode combinée restent à 13 % et 12 %. Pour une charge de travail de 1500 tâches, les techniques DVFS et consolidation permettent une légère amélioration du respect des délais, tandis que la méthode combinée entraîne une légère dégradation. Enfin, à 2000 tâches, presque toutes les tâches ratent

également le respect de délai, quelle que soit la technique utilisée, avec une légère amélioration apportée par le DVFS et la consolidation, tandis que la méthode combinée maintient un niveau similaire à cette dernière.

Les trois techniques d’optimisation ont un impact variable sur le respect de délai. Dans le cas de DVFS, le respect de délai reste acceptable en charge faible et même légèrement meilleur qu’avec la configuration non optimisée dans certains cas, malgré une baisse globale avec l’augmentation de la charge de travail.

La consolidation montre une légère amélioration du respect de délai par rapport au scénario non optimisé à partir de la charge moyenne-élevée, donc plus efficace quand la charge est élevée, même si certains cas de surcharge affectent la stabilité du respect de délai (avec un respect de délai acceptable en charge faible).

La méthode combinée présente un comportement variable selon la charge de travail, entraînant parfois une légère amélioration, et d’autres fois une légère dégradation du respect des délais. De manière générale, l’effet de chaque méthode dépend fortement du niveau de charge de travail appliqué. Un choix judicieux de la technique est essentiel pour maintenir un bon compromis entre consommation énergétique et respect de délai.

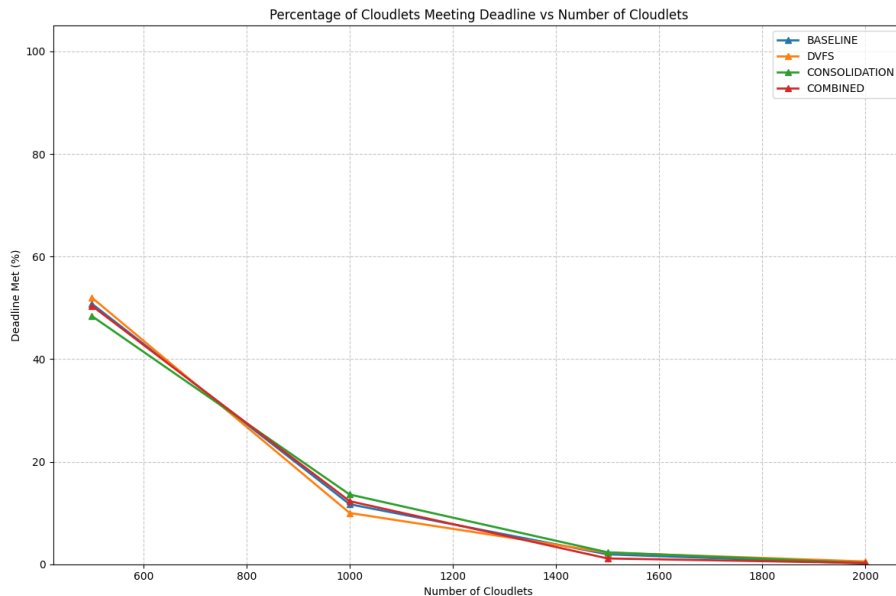


FIGURE 3.3 – Respect de délai par rapport au nombre de tâches.

3.7 Synthèse

Les résultats obtenus à travers les simulations mettent en évidence les bénéfices notables des techniques d’optimisation énergétique appliquées au cloud computing. La technique la plus performante retenue est le DVFS (Dynamic Voltage and Frequency Scaling) appliqué seul ou combiné à la consolidation, en fonction du nombre de tâches traitées. Pour 500, 1000 et 1500 tâches, l’appli-

cation du DVFS seulement a réduit de manière significative la consommation d'énergie, atteignant respectivement 48,49 %, 47,65 % et 52,70 %, par rapport à une méthode n'ayant pas d'optimisation. Pour 2000 tâches, la méthode combinée DVFS + consolidation a permis d'obtenir les résultats les plus pertinents avec un taux de réduction énergétique de 48,07%. En ce qui concerne la qualité de service, l'impact est acceptable : le temps d'exécution global diminue systématiquement avec la mise en œuvre du DVFS ou de la méthode combinée. Cette diminution s'explique par la nature indépendante des tâches, qui rend leur traitement plus fluide et moins contraint, même en présence de techniques d'économie d'énergie. Pour le respect de délai, pour 500 tâches, le taux est légèrement amélioré avec la méthode DVFS. En revanche, pour 1000 tâches un respect des délais légèrement inférieur est observé avec le DVFS. Pour 1500 tâches ce taux augmente légèrement, ce qui montre une légère progression en matière de respect de délais.

Enfin, pour 2000 tâches, la méthode combinée DVFS et consolidation permet une légère hausse du respect de délai, malgré la forte charge de travail. Cela montre que quel que soit le nombre de tâches, l'écart entre la méthode non optimisée et les méthodes de réduction de consommation d'énergie en termes de respect de délai reste relativement faible, ce qui indique que les techniques d'économie peuvent être mises en œuvre sans que cela compromette gravement le respect de délai des tâches.

De manière générale ces résultats indiquent un bon équilibre entre performance et consommation d'énergie, et dans plusieurs cas les techniques appliquées permettent de réduire fortement la consommation tout en conservant, voire en améliorant légèrement, les performances globales du système. Cela étant, cet équilibre n'est pas dans tous les cas, car certains scénarios résultent en une légère dégradation de certains indicateurs de qualité de service comme le respect de délai sous charge moyenne (1000 tâches). Cependant, il est important de noter que ces résultats ne sont pas toujours constants et peuvent varier d'une exécution à une autre, en fonction des conditions initiales, de la répartition des tâches ou du comportement dynamique du simulateur. D'autre part, pour le défi de la scalabilité, en effet il est d'autant plus difficile d'optimiser la gestion des ressources que le nombre de tâches ou de machines à gérer est important. Les techniques d'optimisation doivent être donc adaptatives et capables de se réajuster dynamiquement à l'évolution de la charge de travail pour maintenir les performances tout en garantissant un bon niveau de qualité de service.

3.8 Conclusion

Dans ce chapitre, nous avons présenté notre contribution dans le cadre de ce PFE, nous avons précisé les outils de simulation que nous avons utilisés, le langage Python, l'environnement Visual Studio Code, la bibliothèque SimPy, puis les algorithmes d'optimisation énergétique que nous avons appliqué, mise à l'échelle dynamique de la tension et de la fréquence (DVFS), la consolidation dynamique des machines virtuelles, la combinaison de ces deux techniques, les pseudocodes de chaque algorithme ainsi que les différentes étapes de leur implémentation étaient présentés. Nous

avons exécuté des simulations selon plusieurs scénarios, en variant le nombre de tâches, et les performances ont été évaluées selon trois principales métriques : la consommation d'énergie, le temps d'exécution et le respect de délai.

Les résultats apportés par les différentes simulations effectuées indique que si la stratégie DVFS utilisée seule, donne les meilleurs résultats pour des charges faibles à moyennes, plus la charge de travail est élevée, plus la technique combinée (DVFS + consolidation) montre un comportement performant. Globalement, ces techniques permettent de réduire la consommation d'énergie tout en assurant un bon niveau de qualité de service malgré quelques variations selon le nombre de tâches.

Conclusion générale

Conclusion générale

Le cloud computing constitue une avancée technologique novatrice qui favorise le déploiement des applications dans de multiples domaines d'activité, tout en préservant de la flexibilité, de l'évolutivité et en réalisant des gains économiques. Les enjeux liés cependant à cette technologie sont nombreux notamment la gestion de la consommation énergétique des centres de données la rendant de plus en plus préoccupante en raison des impacts environnementaux et économiques plus larges qu'elle engendre.

Dans ce mémoire, nous avons pris pour objet l'étude des techniques de réduction consommation énergétique déployées dans le cloud computing, plus particulièrement trois techniques majeures dans cette recherche : la mise à l'échelle dynamique de la tension et de la fréquence (DVFS) et la consolidation dynamique de machines virtuelles et la combinaison de ces deux techniques.

Ces techniques cherchent à réduire la consommation d'énergie tout en conservant une qualité de service satisfaisante, en particulier sur le respect des délais et le temps d'exécution. Les simulations effectuées à l'aide de SimPy et Python ont montré l'efficacité de ces techniques, à savoir :

- Le DVFS (Dynamic Voltage Frequency Scaling) permet de réduire la consommation d'énergie en ajustant dynamiquement la fréquence et la tension des processeurs, ces valeurs sont modifiées en continu en fonction des charges de travail à traiter, car celles-ci influent sur la consommation énergétique.
- La consolidation des machines virtuelles rend possible une meilleure utilisation des ressources physiques en accueillant les machines virtuelles sur un nombre restreint de serveurs, tandis que les autres sont sous-utilisées, et doivent être arrêtées (éteintes).
- La combinaison des deux techniques constitue un compromis entre qualité de service et économie d'énergie, adapté aux charges de travail élevées.

Les résultats obtenus montrent que ces méthodes permettent de réaliser des économies d'énergie sans compromettre la qualité de service, qui peut même s'améliorer légèrement, illustrant ainsi un compromis équilibré.

Il conviendrait de s'intéresser à l'utilisation de l'intelligence artificielle pour l'ensemble des décisions d'économie d'énergie pour les études à venir. En effet, des approches telles que l'apprentissage automatique, et l'apprentissage profond (Deep learning), les algorithmes génétiques, les réseaux de neurones et bien sûr le reinforcement learning pour l'optimisation de la consommation d'énergie, pourraient permettre de s'adapter plus facilement aux variations de charge de travail. En particulier, le reinforcement learning permettrait à un système d'apprendre tout seul les meilleures stratégies de réduction de consommation d'énergie selon l'environnement, et ainsi contribuer à optimiser l'énergie consommée de manière dynamique, continue, et plus intelligente.

BIBLIOGRAPHIE

- [1] ANDREW.S.T . (2014) . «*Modern Operating Systems*»(4th ed.) . Pearson Education.
- [2] BENHAMMOUDA.M . (2021) . «*Contribution à optimisation d’ordonnancement de workflows dans un environnement cloud*». Thèse de doctorat. Université de Djallali Liabes - Sidi Bel Abbes.
- [3] BERL.A, GELENBE.E ,DI GIROLAMO.M, GIULIANI.G, DE MEER.H, DANG.MQ, PENTIKOUSIS.K . (2010) . «*Energy-Efficient Cloud Computing*». *The Computer Journal* ,53 (7).
- [4] BHARANY. S . (2022) . «*Une étude systématique sur les techniques économes en énergie dans le cloud computing durable*» . *Durabilité*. 14(6256).
- [5] BOUHARICHE.A, BOUROUIS.H . (2021) . «*Un système basé sur les certificats et les commentaires pour une gestion de confiance dynamique dans un environnement cloud multi-domaines* » . Mémoire de master . Université Mohammed Seddik Ben Yahia - Jijel .
- [6] BOULAUCHE.M, BOUFATIS.D . (2016) . «*Développement d’une interface Web au-tour du module Horizon d’Openstack (Cloud Computing)*» .Mémoire de master. Université de Mou-loud Mammeri - Tizi-Ouzou.
- [7] BOUZIDI.Z . (2019) . «*Pronostic des systèmes industriels basé sur l’intelligence artificielle Maintenance prédictive*» . Thèse de doctorat. Université de Mohamed Khider – Biskra.
- [8] Choukairy.F.Z . (2018) . «*Optimisation de la consommation d’énergie dans un environnement Cloud* » . Mémoire de master .Université de Laval - Canada.
- [9] COTES-RUIZ.I.T, PRADO.R.P, GARCÍA-GALÁN.S, MUÑOZ-EXPÓSITO.J.E, RUIZ REYES .N . (2017) . «*Dynamic Voltage Frequency Scaling Simulator for Real Workflows Energy-Aware Management in Green Cloud Computing* ». PLoS ONE, 12 (1), e0169803.
- [10] EL GHOUBACH.I . (2020) . «*La Sécurité des Données dans le Cloud Computing*» . Thèse de doctorat. Université de Sidi Mohamed Ben Abdellah - Fes.
- [11] Fabien.D . (2015) . «*Optimisation énergétique de data centers par utilisation de liquides pour*

- le refroidissement des baies informatiques* ». Thèse de doctorat. Université de Bretagne Sud - France.
- [12] FEMMAM. M . (2018). « *Une approche formelle pour la planification des tâches pour la QoS dans le cloud-computing* ». Thèse de doctorat. Université de Mohamed Khider - Biskra.
- [13] GHADDAR.A . (2013). « *Une contribution à la gestion des applications SaaS mutualisées dans le cloud : approche par externalisation* ». Thèse de doctorat. Université de Nantes - France.
- [14] GILL.S.S , BUYYA.R . (2017). « *A Taxonomy and Future Directions for Sustainable Cloud Computing :360 Degree View* ». arXiv preprint arXiv :1712.02899, 2017.
- [15] HAMDANI.N, KERROUM.S, OUALLOUCHE.N . (2019). « *Etude et comparaison des failles de sécurité d'OpenStack et OpenNebula* ». Mémoire de master . Université de Mouloud Mammeri - Tizi-Ouzou.
- [16] IBRAHIM.S . (2015). « *Optimisation d'Infrastructures de Cloud Computing dans des Green Datacenters* ». Thèse de doctorat. Ecole Doctorale Mathématiques, Sciences et Technologies de l'Information, Informatique (MSTII) . Laboratoire d'Informatique de Grenoble (LIG) - France.
- [17] ILAGER.S, BUYYA.R . (2021). « *Energy and Thermal-aware Resource Management of Cloud Data Centres : A Taxonomy and Future Directions* ». arXiv preprint arXiv :2107.02342.
- [18] KACED.Y . (2018). « *Etudes du refroidissement par free cooling indirect d'un bâtiment exothermique : Application au centre de données* ». Thèse de doctorat. Université de Bretagne Sud - France.
- [19] Kella.A . (2015). « *Vers une gestion de migration des machines virtuelles pour améliorer la qualité de service et réduire la consommation d'énergie dans les cloud computing.* ». Thèse de doctorat. Université de Ahmed Ben Bella - Oran.
- [20] LI.G, JIANG.Y, YANG.W, HUANG.C . (2016). « *Self-Adaptive Consolidation of Virtual Machines For Energy-Efficiency in the Cloud* ». arXiv preprint arXiv :1604.04482.
- [21] MAMMERI.A . (2014). « *Amélioration des performances énergétiques des systèmes de refroidissement industriels : Application aux serveurs informatiques* ». École Nationale Supérieure d'Arts et Métiers Spécialité " Énergétique " - Paris .
- [22] MANIRAKIZA.N.M . (2017). « *La migration en temp réel des machines virtuelles sans le cloud computing* ». Memoire de Master. Université de Mouloud Mammeri - Tizi-Ouzou.
- [23] MATERWALA.H, ISMAIL.L . (2022). « *Performance and energy-aware bi-objective tasks scheduling for cloud data centers* ». Procedia Computer Science, 197, pp. 238–246.
- [24] MEDHIOUB.H . (2015). « *Architectures et mécanismes de fédération dans les environnements Cloud Computing et Cloud Networking* ». Thèse de doctorat. Université de Pierre et Marie Curie - Paris.
- [25] MOHAMMEDI.F . (2024). « *Harris Hawks Optimization Algorithm (HHO) pour l'ordonnement des tâches dans le Cloud Computing* ». Mémoire de master. Université Abou Bakr Belkaid - Tlemcen.

- [26] PATTERSON.M, TSCHUDI.B and all . (2010) . « *ERE : A Metric for Measuring the Benefit of Reuse Energy from a Data Center* » . *White Paper #29, The Green Grid*.
- [27] SHARMA.Y, JAVADI.B, SI.W, SUN.D . (2016) . « *Reliability and energy efficiency in Cloud Computing Systems : survey and taxonomy* » . *Journal of Network and Computer Applications*.
- [28] TSENG.C-Y, LEE.L-T, LIU.K-Y, HUANG.H-Y . (2013) . « *A Dynamic Resource Management with Energy Saving Mechanism for Supporting Cloud Computing* » . *International Journal of Grid and Distributed Computing*, 6 (1).
- [29] USMAN.M.J and All . (2019) . « *Energy-Efficient Nature-Inspired Techniques in Cloud Computing Datacenters* » . *Telecommunication Systems*, 71(2).
- [30] VARASTE.H.A, GOUDARZI.M . (2017) . « *Server Consolidation Techniques in Virtualized Data Centers : A Survey* » . *IEEE Systems Journal*, 11 (2).
- [31] VERDUN.G, AZEVEDO.D and all . (2008) . « *The Green Grid Metrics : Data Center Infrastructure Efficiency (DCiE) Detailed Analysis* » . *White Paper #14, The Green Grid*.
- [32] Wang.S, Qian.Y, Yuan.J, and You.I . (2017) . « *A DVFS based energy-efficient tasks scheduling in a data center* » . *IEEE Access*, vol. 5, pp. 13090–13102 .
- [33] Zhang.Y, Wei.Z and Zhang.M . (2017) . « *Free cooling technologies for data centers : Energy saving mechanism and applications.* » . *Energy Procedia*, vol. 143, pp. 410–415.
- [34] ZHOU.Q, XU.M, GILL.SS, GAO.C, TIAN.W, XU.C, BUYYA.R . (2020) . « *Energy efficient algorithms based on Vm consolidation for cloud computing : comparisons and evaluations* » . *arXiv preprint arXiv :2002.04860*.
- [35] *Cloud broker*. Wikipedia. Disponible à : https://en.wikipedia.org/wiki/Cloud_broker.(Consulté le Janvier 2025).
- [36] *Cloud computing*. Wikipedia. Disponible à : https://fr.wikipedia.org/wiki/Cloud_computing.(Consulté le Janvier 2025).
- [37] *Définition Visual Studio Code*. Bility, s.d. [En ligne]. Disponible à : <https://bility.fr/definition-visual-studio-code>.(Consulté le Mai 2025).
- [38] *Fournisseurs Cloud Computing : Quel SaaS choisir pour son infrastructure cloud ?*. Antoine Grignola, 15 juillet 2024. [En ligne]. Disponible à : <https://www.data-bird.co/blog/fournisseurs-cloud-computing>. (Consulté le Janvier 2025).
- [39] *Fournisseurs de cloud : le comparatif des meilleures solutions en 2025*. Axido, 2022. [En ligne]. Disponible à : <https://www.axido.fr/fournisseurs-de-cloud-le-comparatif/>. (consulté en Janvier 2025).
- [40] *L'appétit vorace des centres de données inquiète le Département de l'Énergie US. Berkeley Lab (USA), 22 Janvier2025*. [En ligne]. Disponible à : <https://www.enerzine.com/lappetit-vorace-des-centres-de-donnees-inquiete-le-departement-de-lenergie-us/154840-2025-01>. (Consulté le Février 2025).

- [41] *Le module enum – Le langage Python*. HEIG-ARC, s.d. [En ligne]. Disponible à : <https://he-arc.github.io/livre-python/enum/index.html#:~:text=D%C3%A9finition,code%20plus%20lisible%20et%20humain>.(Consulté le Mai 2025).
- [42] *Le module random – Le langage Python*. HEIG-ARC, s.d. [En ligne]. Disponible à : <https://he-arc.github.io/livre-python/random/index.html>.(Consulté le Mai 2025).
- [43] *Pourquoi et comment utiliser Matplotlib avec Python*. Wild Code School, s.d. [En ligne]. Disponible à : <https://www.wildcodeschool.com/blog/pourquoi-et-comment-utiliser-matplotlib-avec-python> .(Consulté le Mai 2025).
- [44] *SimPy Documentation*. Read the Docs, s.d. [En ligne]. Disponible à : <https://simpy.readthedocs.io/en/latest/>.(Consulté le Mai 2025).
- [45] *Téléchargement Python – 01net*. 01net.,s.d. [En ligne]. Disponible à : <https://www.01net.com/telecharger/programmation/creation/python.html>. (Consulté le Mai 2025).
- [46] *Time — Time access and conversions – Python documentation*. Python Software Foundation, s.d. [En ligne]. Disponible à : <https://docs.python.org/3/library/time.html> .(Consulté le Mai 2025).
- [47] *Tqdm en Python : une barre de progression pour vos boucles*. DataCamp, s.d. [En ligne]. Disponible à : <https://www.datacamp.com/fr/tutorial/tqdm-python> .(Consulté le Mai 2025).
- [48] *Virtualisation*. IBM France. [En ligne]. Disponible à : <https://www.ibm.com/fr-fr/topics/virtualization>. (Consulté le Février 2025).
- [49] *Virtual Machines (VMs) and Their Role in Cloud Computing*. Zulfi Al Hakim, 29 novembre 2024. [En ligne]. Disponible à : <https://btech.id/en/news/virtual-machines-vms-and-their-role-in-cloud-computing/>.(Consulté le Janvier 2025).