

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Saad Dahlab Blida



Faculté des sciences

Département informatique

Mémoire de fin d'études

En vue d'obtenir le diplôme de master

Filière : Informatique

Spécialité : Informatique

Option : Ingénierie des logiciels

THEME :
Vers un système auto-adaptatif pour le
E-Learning

Réalisé par:

Hout Hounaida

Grini Hadda

Soutenu le : 24septembre 2022, Devant le Jury compose de :

Présidente Mme. I. CHIKHI.

Examinatrice Mme. I. FERDI.

Promotrice Mme. D. GUESSOUM.

Promotion : 2021/2022

ملخص

تقوم أنظمة التكيف الذاتي بتعديل هيكلها وسلوكها استجابة للتغيرات في البيئة.

في إطار هذا المشروع، اهتمنا باستخدام التكيف الذاتي في مجال التعلم الإلكتروني بسبب أهميته وميزته، خاصة مع ظهور فيروس كورونا. لقد اقترحنا طريقة تستند الأدلة العليا (خوارزمية التحسين بواسطة أسراب معينة) والتي تسمح لمنصة التعلم الإلكتروني بتكييف سلوكها من خلال تغيير وظائفها (الدروس والتصميم والترجمة وما إلى ذلك) وفقاً لسرعة اتصال المستخدم ومع مراعاة وقت الاستجابة وكذلك استهلاك البيانات لاتصال المنصة. تم إجراء العديد من الاختبارات باستخدام منصة التعلم الإلكتروني التي صممناها لتقييم فعالية الإستراتيجية المقترحة.

الكلمات المفتاحية: التكيف الذاتي، التعلم الإلكتروني، الأدلة العليا، خوارزمية التحسين بواسطة أسراب معينة.

Résumé

Les systèmes auto-adaptatifs modifient leur structure et leur comportement en réponse aux changements de l'environnement.

Dans le cadre de ce projet, nous nous sommes intéressés à l'utilisation de l'auto-adaptation dans le domaine du e-Learning en raison de sa pertinence et de son avantage, notamment avec l'avènement de COVID-19. Nous avons proposé une méthode basée sur un méta heuristique (algorithme d'optimisation par des essais particuliers) qui permet à la plateforme E-Learning d'adapter son comportement en variant ses fonctionnalités (cours, design, Traduction, etc.) en fonction de la vitesse de connexion de l'utilisateur et en prenant en compte le temps de réponse ainsi que la consommation de données de la connexion de la plateforme. Plusieurs tests ont été réalisés à l'aide d'une plateforme E-Learning que nous avons conçue afin d'évaluer l'efficacité de la stratégie proposée.

Mots clés : Auto adaptation, E-learning, métaheuristique, Algorithme d'optimisation par des essais particuliers.

Abstract

Self-adaptive systems change their structure and behavior in response to the unpredictability of the environment.

In this project, we are interested in the use of self-adaptation in the e-learning domain because of its relevance and advantage, especially with the advent of COVID-19. We have proposed a method based on a meta heuristic (optimization algorithm by particular swarms) that allows the E-learning platform to adapt its behavior by varying its functionalities (courses, design, translation, etc.) according to the user's connection speed and taking into account the response time as well as the data consumption of the platform connection. Several tests were carried out using an E-learning platform that we designed in order to evaluate the effectiveness of the proposed strategy.

Keywords: Self-adaptation, E-learning, Meta heuristics, optimization algorithm by particular swarms.

Remerciement

Nous tenons à la fin de ce travail à remercier ALLAH le tout puissant de nous avoir donné la puissance, foi et de nous avoir permis d'en arriver là.

Nos vifs remerciements accompagnés de toute notre gratitude vont également à notre promotrice Mme D. Guessoum. Pour son soutien, sa disponibilité, ses précieux conseils, sa confiance, ses orientations et ses encouragements.

Nous tenons aussi à exprimer notre gratitude à nos profs qui nous ont formés durant notre cycle universitaire, ainsi que tout le staff du département d'informatique

Nous remercions les membres de jury Mme I. CHIKHI et Mme I. FERDI, de nous avoir fait l'honneur d'accepter de juger ce modeste travail. Nos remerciements vont également à nos parents, de tous les sacrifices qu'ils ont consentis pour nous permettre de suivre nos études dans les meilleures conditions possibles et n'avoir jamais cessé de nous encourager tout au long de nos années d'étude.

Nous remercions tous ceux qui nous ont aidés à l'aboutissement de ce travail.

Enfin, nous espérons que ce mémoire servira d'exemple et de support pour les années à venir.

Dédicace

*Je dédie ce modeste travail A mes chers parents qui m'ont tout donné
durant ma vie, et tous les mots de l'univers ne suffisent pas pour les
remercier,*

*A mes sœurs. Pour leur Soutien et encouragement. A mon binôme
Hadda,*

A mon amis Akila,

*A toute ma famille pour leur soutien tout au long de mon parcours
universitaire,*

A mes enseignants et tous les enseignants,

A toute la promotion 2021-2022.

Hounaïda

Dédicace

*Je dédie ce modeste travail A mes chers parents qui m'ont tout donné
durant ma vie, et tous les mots de l'univers ne suffisent pas pour les
remercier.*

A mes sœurs et mes frères. Pour leur Soutien et encouragement.

A mon binôme Hounaïda

A mon amis Akila, Khadidja et khaoula.

A mes enseignants et tous les enseignants.

A tout qu'ils sont proche de moi.

Hadda

Table des matières

Introduction générale	1
Problématique	1
Objectif	1
CHAPITRE 1 :	3
Généralités sur l'auto-adaptation des systèmes logiciel.....	3
1 Introduction.....	4
1 Définition de l'auto adaptation	4
3 L'intérêt des systèmes auto adaptatifs.....	5
4 Propriétés des systèmes auto-adaptatifs	6
4.1. Auto-configuration.....	6
4.2. Auto-optimisation	7
4.3 Auto réparation	7
4.4 Auto protection	7
5 Trois générations de systèmes auto-adaptatifs	7
6 Les types d'un système auto-adaptatif	8
7 Conception de système adaptatif.....	9
8 Les éléments d'un système qui peuvent porter une adaptation.....	9
9 Boucle de contrôle	10
10 Méthodologies existantes.....	10
10.1 Technologie de contrôle.....	11
10.2 Programmation logicielle.....	11
10.3 Intelligence artificielle :	12
11 Les défis des systèmes auto-adaptatifs.....	12
11.1 Défis de confiance.....	12
11.2 Défis d'analyse	12
11.3 Défis de planification.....	13
11.4 Défis d'exécution.....	13
12 Systèmes existants représentatifs	14
13 Bilan de ces recherches	16
14 Conclusion	17
CHAPITRE 2 :	18

Etat de l'art sur E-Learning	18
1 Introduction.....	19
2 Evolution de E-learning	19
3 Definition « E-Learning ».....	20
4 Les avantages et les inconvénients.....	22
4.1 Les avantages	22
4.2 Les inconvénients.....	23
5 Quelques plateformes E-Learning	24
6 L'adaptation dans le E-Learning	26
6.1 L'adaptation de la navigation :	27
6.2 Adaptation du contenu :	27
6.3 Adaptation de la présentation :	28
7 Les plateformes E-Learning adaptatives :	28
8 Discussion :	32
9 Conclusion :	33
CHAPITRE 3 :	34
Algorithme d'optimisation par les essais particuliers PSO.	34
1 Introduction.....	35
2 Problème d'optimisation.....	35
3 Algorithmes de résolutions	35
3.1 Les méthodes exactes.....	36
3.2 Les méthodes approchées.....	36
4 Classification des algorithmes de résolution :	37
5 Fonction Objectif :	38
6 Algorithme Optimisation par les essais particulaire (PSO)	38
6.1 Principe générale :	38
6.2 Formalisation Mathématique :	40
6.3 Principe de fonctionnement de l'algorithme PSO :	41
6.4 Avantages de PSO:	43
6.5 Inconvénients de PSO: [28]	43
7 Conclusion	44
CHAPITRE 4 :	45
Conception et spécification de l'approche.	45

1	Introduction :	46
2	Etude de cas :	46
3	Modélisation des variations dans la plateforme proposée.....	47
4	Modélisation de notre système.....	48
4.1	Diagramme des cas d'utilisation	48
4.2	Diagramme de classe	49
5	Description de l'approche	50
6	Déclenchement de notre approche	51
7	Fonctionnement de l'approche.....	53
7.1	But de notre approche	54
7.2	Tableau de variations	54
7.3	Spécification de l'algorithme PSO pour notre approche.....	56
7.4	Codage de l'essaim particulière	57
7.5	Descriptions des services :	57
7.6	La population initiale :	60
7.7	La fonction de fitness.....	61
7.8	Matrice de limites :	62
7.9	La fonction de fitness normalisée :	62
7.10	Exemple applicatif :	63
7.11	Les étapes de l'algorithme PSO :.....	63
8	Conclusion :	66
	CHAPITRE 5 :	66
	Implémentation de la plateforme et test.	66
1	Introduction :	67
2	Les outils utilisés :	67
2.1	Les Systèmes de Gestion de Contenu (CMS) :	67
2.2	L'environnement de développement Apache Netbeans IDE 14 :	70
2.3	Highcharts	70
3	Présentation de notre plateforme.....	70
3.1	Présentation les services (fonctionnalités) de notre plateforme :	71
4	Présentation de la simulation du l'approche	76
5	Expérimentations et résultats	78

6	Exemple de travail :	78
6.1	Simulation avec PSO :	78
6.2	Justification des paramètres :	79
6.3	Etat d'essaim initial :	81
6.4	Simulation sans PSO :	82
7	Temps de réponse	83
8	Consommation de connexion	85
9	Utilité :	86
10	Temps d'exécution d'approche :	87
11	Comparaison entre l'algorithme génétique (GA) et PSO :	87
12	Analyse des graphes :	88
13	Discussion :	89
14	Conclusion :	89
	Conclusion générale	67
	Bibliographie	67

Liste des figures

Figure 1: Propriétés des systèmes auto-adaptatifs.	6
Figure 2 : Boucle de contrôle.	10
Figure 3: Courbe de difficulté du maintien de la température ambiante.	15
Figure 4 : Exemple représentatif d'un Astrolab.	16
Figure 5 : Structure du e-Learning.	20
Figure 6: quelques termes souvent associés au E -Learning.	21
Figure 7 : une simple interface de Moodle.	24
Figure 8: une simple interface d'ATutor.	25
Figure 9: une simple interface de Dokeos.	25
Figure 10: une simple interface de Coursera.	26
Figure 11: une simple interface d'AHA.	29
Figure 12: une simple interface d'ELM-ART	29
Figure 13: une simple interface d'iWeaver.	30
Figure 14: une simple interface de NetCoach.	31
Figure 15: Taxonomie des méthodes de résolution des problèmes d'optimisation.	37
Figure 16 : (a) Nuage d'oiseaux (b) bancs de poissons .	38
Figure 17 : Déplacement d'une particule.	40

Figure 18 : fonctionnement de l’algorithme PSO	42
Figure 19 : Organigramme de principe de PSO.	43
Figure 20: Modélisation des variations	47
Figure 21:Diagramme de cas d'utilisation.	49
Figure 22 : Diagramme de classe	50
Figure 23:Représentation de latence.	51
Figure 24: Interface de Local by Flywheel	68
Figure 25: Interface des thèmes WordPress	69
Figure 26 : Interface des extensions WordPress.	69
Figure 27 : Interface d’authentification	71
Figure 28 : Interface d’accueil	72
Figure 29: Interface de recherche.	72
Figure 30 : Interface de cours	73
Figure 31 : les avis de cours	73
Figure 32 : service traduction	74
Figure 33 : service de communication	74
Figure 34 : services désigne	75
Figure 35 : Interface d'accueil de simulation	76
Figure 36 : Interface de paramètres d’algorithme PSO	77
Figure 37 : Interface de paramètre sans PSO	77
Figure 38 : évolution de Gbest	80
Figure 39 : journal de simulation avec PSO	82
Figure 40 : journal de simulation sans PSO	83
Figure 41 : Teste de temps de réponse	84
Figure 42 : Test 1 de consommation des connexions	85
Figure 43 : Test d’utilité	86
Figure 44 : comparaison du temps de réponse entre GA et PSO	87
Figure 45: comparaison d’utilité entre GA et PSO	88
Figure 46: comparaison de consommation entre GA et PSO.	88

Liste des tableaux

Tableau 2 : Les différentes adaptations effectuées sur les plateformes adaptatives.	32
Tableau 3:Tableau de variations	55
Tableau 4:Tableau de variations	56
Tableau 5:Codage des états.	57
Tableau 6:Codage de service de cours	58

Tableau 7:Codage de service de Design _____	58
Tableau 8:Codage de service de Navigation_Liens _____	58
Tableau 9:Codage de service de Message _____	58
Tableau 10:Codage de service de Téléchargement _____	58
Tableau 11:Codage de service de Police _____	59
Tableau 12:Codage de service de Statistique _____	59
Tableau 13:Codage de service de Commentaire _____	59
Tableau 14:Codage de service de Streaming vidéo (Qualité) _____	59
Tableau 15:Codage de service de Traduction. _____	59
Tableau 16:Codage de service de Quiz. _____	60
Tableau 17:Exemple d'une population _____	60
Tableau 18:les poids des critères _____	62
Tableau 19 :Les paramètres initiaux _____	63
Tableau 20:Fitness des particules _____	64
Tableau 21: paramètre de simulation avec PSO _____	78
Tableau 22 : Test de nombre d'essaim _____	79
Tableau 23 : état initial d'essaim _____	81
Tableau 24 : paramètre de simulation sans PSO. _____	83

Abréviation :

SAA : Système Auto Adaptatif.

MAPE-K: Monitor-Analyze-Plan-Execute over a shared Knowledge.

IBM: International Business Machines.

LAN: local area network.

IHM: Interface Homme-Machine.

AHA Adaptative Hypermedia architecture.

ELM-ART: Episodic Learner -Model Adaptative Remote Tutor.

Lisp: List Processing.

PSO: Particle Swarm Optimization.

Pbest: Personal Best.

Gbest: Global Best.

UML: Unified Modeling Language.

Uc: Use case.

MS: Milli Second.

Kbps: Kilobit par second.

APSO: Algorithm Particle Swarm Optimization.

ML: Matric Limite.

CMS: Content Management System.

PHP: Hypertext Preprocessor.

HTML: Hyper Text Markup Language.

MySQL: My Structured Query Language.

IDE: Integrated Development Environment.

RAM: Random Access Memory.

GA : Génétic Algorithm

Introduction générale

En raison de la complexité des systèmes logiciels actuels, la communauté du génie logiciel s'est penchée sur de nouvelles façons de concevoir, de déployer, de maintenir et d'améliorer les systèmes et services à forte composante logicielle. En plus de s'adapter aux changements de leurs contextes d'exploitation, de leur environnement et de leurs besoins, les systèmes logiciels doivent devenir plus adaptables, flexibles, robustes, fiables, économes en énergie, récupérables, personnalisables, configurables et auto-optimisables. Par conséquent, l'auto-adaptation des systèmes qui peuvent modifier leur comportement et/ou leur structure en réponse à leur perception de l'environnement et du système lui-même, ainsi que de ses exigences est apparue comme une question d'étude importante dans un large éventail de domaines d'application.

Problématique

Les plates-formes d'apprentissage en ligne contiennent plusieurs services et technologies qui facilitent le processus d'apprentissage et tentent de remplacer les écoles. Cependant, dans certains pays, tous les utilisateurs ne disposent pas d'une bonne vitesse de connexion, il est donc nécessaire de trouver une méthode ou une approche qui permette aux plates-formes d'apprentissage en ligne de s'adapter aux faibles vitesses de connexion des utilisateurs et de commercialiser avec un bon temps de réponse.

Les plates-formes, quant à elles, doivent adapter leurs fonctionnalités à l'évolution constante des ressources et des demandes des utilisateurs. Cependant, le temps est crucial pour un service d'adaptation, car les plates-formes doivent être converties sans que le temps supplémentaire consacré au processus d'adaptation ne soit reconnu.

Objectif

Ce PFE vise à développer une méthode alternative pour générer automatiquement des plans d'adaptation pendant l'exécution. La méthode proposée doit développer des plans d'adaptation optimaux basés sur plusieurs paramètres tels que le temps de réaction, l'utilité et l'utilisation des données de connexion. Ceci devrait être accompli en employant des métaheuristiques (algorithme d'optimisation par les essais particuliers SO) La technique suggérée doit être testée à l'aide d'une plateforme d'apprentissage en ligne afin de démontrer qu'elle est efficace et qu'elle s'adapte aux mauvaises connexions.

Introduction générale

Le mémoire est structuré en cinq chapitres, suivis d'une conclusion générale :

Le premier chapitre : Généralités sur l'auto-adaptation des systèmes logiciels : est consacré à la présentation des concepts de base des systèmes auto adaptatifs.

Le deuxième chapitre : Etat de l'art sur E-Learning : ce chapitre présente l'importance de E-Learning.

Le troisième chapitre : Algorithme d'Optimisation par les essais particuliers SO : nous présentons dans ce chapitre le concept de métaheuristique, ainsi que ses propriétés et classifications. La métaheuristique "algorithme d'optimisation par les essais particuliers PSO" utilisée dans notre étude est ensuite présentée.

Le quatrième chapitre : Conception et spécification de l'approche : nous présentons la modélisation de notre plateforme E-Learning ainsi que la conception de notre solution basée sur l'algorithme PSO et sa spécification.

Le cinquième chapitre : Implémentation de la plateforme et test : ce dernier chapitre de ce mémoire présente l'implémentation et les résultats des tests obtenus de notre approche.

Enfin, le mémoire se termine par une **conclusion générale**

CHAPITRE 1 :
**Généralités sur l'auto-adaptation
des systèmes logiciels**

1 Introduction

Dès les débuts de l'informatique, les théoriciens ont reconnu que l'un des aspects les plus frappants de l'informatique est sa capacité potentielle à se modifier : plutôt que de présenter aux utilisateurs un ensemble fixe de calculs déterminés au moment du déploiement, le système pourrait, au moment de l'exécution, modifier à la fois ce qu'il calcule et la manière dont il le calcule. Cependant, bien que l'"auto-modification" soit théoriquement intrigante, peu de systèmes de programmation et de méthodes d'ingénierie l'ont adoptée- les avantages d'une telle démarche n'étaient pas évidents étant donné la complexité supplémentaire du raisonnement sur le comportement du système et le risque de faire des dégâts par inadvertance.[1]

L'objectif de ce chapitre est de présenter les concepts de bases des systèmes auto adaptatifs.

1 Définition de l'auto adaptation

L'auto-adaptation fournit une méthode prometteuse pour gérer la complexité des systèmes d'aujourd'hui. Un défi mondial dans le développement des Systèmes Auto-Adaptatifs SAA est de savoir comment exprimer les normes afin que le SAA puisse traiter les problèmes posés par le domaine d'application, y compris l'incertitude comportementale.[2]

Afin de clarifier le concept de SAA, nous passons en revue quelques définitions :

Définition 1 :

Le système auto-adaptatif est un système en boucle fermée, qui peut changer lui-même en raison des changements continus du système. Ses exigences et les tendances existantes dans le développement et le déploiement de systèmes complexes réduisent l'interaction homme-machine. La conception d'un système adaptatif dépend des besoins des utilisateurs et des caractéristiques environnementales du système. Les logiciels adaptatifs nécessitent une fiabilité, une robustesse, une adaptabilité et une disponibilité extrêmement élevées .[2]

Définition 2 :

Le logiciel adaptatif modifie son propre comportement en fonction des changements dans son environnement d'exploitation.[3]

Définition 3 :

Lorsque l'évaluation montre qu'il ne fait pas ce que le logiciel est censé faire, ou lorsqu'une meilleure fonctionnalité ou performance est possible, le système adaptatif évalue son propre comportement et modifie ses propres performances.[4]

Définition 4 :

Les systèmes adaptatifs sont également appelés systèmes adaptatifs complexes par la communauté anglo-saxonne.[5] Actuellement, il existe plusieurs définitions dans la communauté :

Définition 4.1 :

Un système adaptatif complexe est un réseau dynamique d'agents (qui peuvent représenter des cellules, des espèces, des individus, des nations) qui fonctionnent constamment en parallèle et répondent aux actions d'autres agents. Le contrôle des systèmes adaptatifs complexes est décentralisé. S'il n'y a pas de comportement cohérent dans le système, il entre en compétition et coopère entre les agents. Le comportement global du système est le résultat d'un grand nombre de décisions que l'agent prend individuellement à chaque instant.[5]

Définition 4.2 :

Un système adaptatif complexe se développe en respectant trois principes clés : l'ordre est différent du prédéterminé, l'histoire du système est irréversible et le comportement futur du système est imprévisible. Les éléments constitutifs des systèmes adaptatifs complexes sont les agents. Les agents scannent leur environnement et développent un schéma de représentation pour l'interpréter, puis créent des règles d'action. Ces scénarios changeront et évolueront avec le temps.[5]

3 L'intérêt des systèmes auto adaptatifs

Le système adaptatif suit les changements fréquents dans l'environnement pour assurer la meilleure qualité de service pour les utilisateurs finaux. Ces systèmes sont gérés de manière autonome. L'objectif inhérent d'un système adaptatif est de maintenir la qualité des services fournis, quel que soit l'état de l'environnement. Il s'agit de stabiliser la qualité de service.[6]

Ce système a trois capacités de base : observation, prise de décision et intro-action :

- **Observation :** Un système adaptatif est un système qui observe son environnement et détecte les changements. D'un point de vue technique, cette capacité se traduit par la présence de sondes (logicielles ou matérielles) pour mesurer les propriétés pertinentes de l'environnement.[6]

- **Prise de décision** : dépend de l'état de l'environnement, mais aussi de son configuration actuelle (capacité d'introspection), le système adaptatif décide seul de la nouvelle configuration à adopter pour fonctionner au mieux.[6]
- **Intro-action** : Afin de modifier sa propre configuration, la manipulation adaptative du système et modifier les éléments qui le composent. Ces " intro-actions " sont des actions de l'intérieur et sont une sorte de capacité d'autoréflexion active.[6]

4 Propriétés des systèmes auto-adaptatifs

Les propriétés du système auto-adaptatif peuvent être résumées en quatre objectifs : auto-configurant, auto-réparateur, auto-optimisant et auto-protecteur, comme le montre la figure ci-dessous. Le système auto-adaptatif peut s'auto-configurer au moment de l'exécution en réponse à l'environnement d'exploitation changeant, à l'auto-ajustement pour optimiser les performances, à l'auto-guérison rencontré des obstacles inattendus pendant le fonctionnement. L'équipe de recherche et développements concentrera sur la construction d'auto-configuration, d'auto-optimisation et le développement de la théorie des systèmes d'autoprotection, des méthodes, des outils et des techniques.[7]

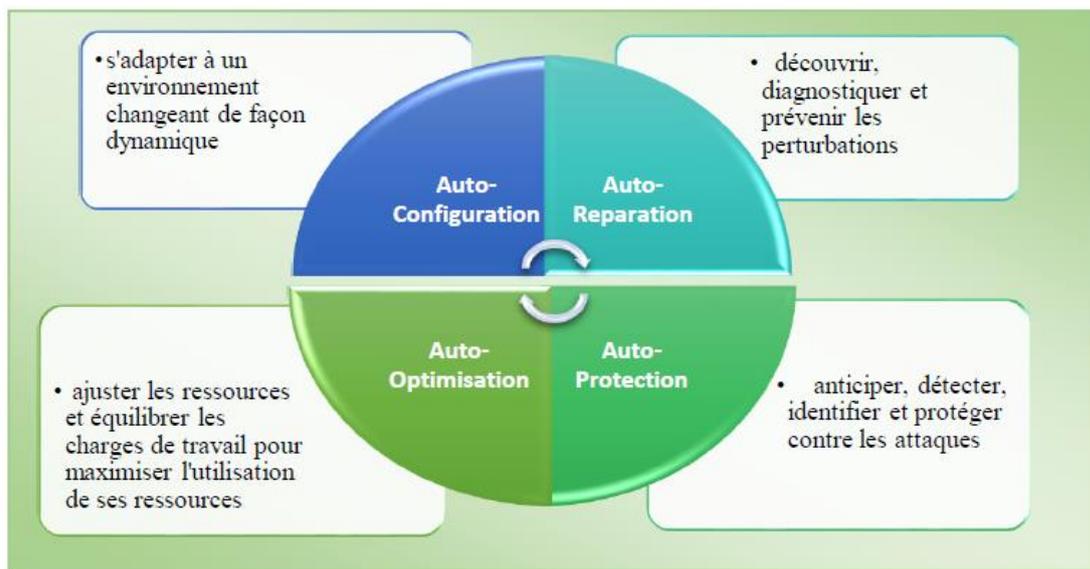


Figure 1: Propriétés des systèmes auto-adaptatifs. [6]

4.1. Auto-configuration

Les systèmes d'Auto-configuration offrent une plus grande réactivité en s'adaptant aux environnements en évolution dynamique. Un système à auto-configurable doit être capable de se configurer et de se reconfigurer dans des conditions changeantes et imprévisibles. Différents

niveaux de participation de l'utilisateur final devraient être autorisés, de la reconfiguration basée sur l'utilisateur à la reconfiguration automatique basée sur des boucles de surveillance et de rétroaction. L'algorithme adaptatif peut apprendre la meilleure configuration pour atteindre les performances requises ou répondre à toute autre exigence fonctionnelle ou non fonctionnelle requise.[7]

4.2. Auto-optimisation

L'auto-optimisation est la capacité d'allouer et d'utiliser efficacement les ressources au maximum pour répondre aux besoins des différents utilisateurs. L'utilisation des ressources et la charge de travail de gestion sont deux aspects importants nécessaires pour cette fonctionnalité. L'un des modèles couramment utilisés dans l'utilisation des ressources est la fonction d'utilité. Les technologies existantes de gestion de la charge de travail, telles que le partitionnement logique et les clusters de serveurs dynamiques, doivent être évolutives vers des systèmes hétérogènes.[7]

4.3 Auto réparation

Les systèmes d'auto réparation offrent une résilience en détectant et en empêchant les pannes et en récupérant les pannes. Un tel système sera en mesure de récupérer sans perte de données ni retard de traitement important en raison d'événements courants et anormaux qui peuvent empêcher certaines parties de fonctionner correctement. L'auto-récupération signifie que le système sélectionnera une configuration alternative pour la configuration actuellement utilisée sous la saisie de l'utilisateur, et passera à cette configuration avec une perte d'informations ou un délai minimal. L'objectif principal de l'auto réparation est de maximiser la disponibilité, la maintenabilité et la fiabilité du système.[7]

4.4 Auto protection

Le système d'autoprotection protège les informations et les ressources en prédisant, en détectant et en se défendant contre les attaques. Un tel système sera capable de se protéger en utilisant la reconnaissance de formes et d'autres technologies pour détecter et arrêter les menaces.[7]

5 Trois générations de systèmes auto-adaptatifs

L'anticipation est l'un des principes de base du génie logiciel. Concevoir un système est en fait, prédisez toutes les situations auxquelles le système devra faire face et prédisez le comportement relatif du système. En d'autres termes, un bon concepteur doit être capable de prévoir les changements dans l'environnement et la configuration du système. Cependant, dans le cas d'un

système adaptatif simple, la complexité de l'environnement produira un grand nombre de configurations possibles, difficiles à prévoir et coûteuses à concevoir individuellement. Par conséquent, la situation idéale est de concevoir un système qui déduit la configuration à utiliser en fonction des changements dans l'environnement. Plus généralement, il est difficile de prédire les états possibles de l'environnement, et un système idéal doit également déduire les états critiques de son environnement, et peut même déduire comment les mesurer.[8]

Ce problème attendu reflète l'existence de trois générations de systèmes potentiels d'auto-adaptation, allant des systèmes statiques aux systèmes, peut déduire la configuration appropriée et les mesures pertinentes à effectuer sur l'environnement.[8]

1. Le système de première génération : est statique ou pseudo-adaptatif. L'évolution et l'architecture des systèmes statiques n'évoluent pas, tandis que l'évolution des systèmes pseudo-statiques supporte certaines évolutions réalisées de manière particulière. En raison du manque d'outils et de technologies appropriés, la plupart des systèmes temps réel sont aujourd'hui conçus de cette manière.[8]

2. Le système de deuxième génération est capable de déduire l'architecture la plus appropriée à l'état de son environnement. Pour cela, ils ont intégré de nombreux capteurs prédéfinis qui leur permettent de détecter les changements dans l'environnement. Certains systèmes mobiles commencent tout juste à intégrer ce type de fonction.[8]

3. Le système de troisième génération peut découvrir de nombreux capteurs disponibles dans son environnement et apprendre à les utiliser pour s'adapter au mieux à son architecture afin d'atteindre divers objectifs prédéfinis. Cette vision liée à l'informatique autonome est une vision que les systèmes multi-agents, les systèmes contextuels, les systèmes intelligents, etc. tendent à réaliser.[8]

6 Les types d'un système auto-adaptatif

Un système intelligent appartient au :

- a. Premier type** d'auto-adaptation s'il possède une sorte d'entité intelligente, le système adaptatif composé des éléments nécessaires pour connaître l'environnement et agir sur lui.[9]
- b. Le deuxième type** requiert principalement un moteur de prise de conscience qualifié pour connaître le modèle au moment de l'exécution du système et pour changer ou influencer la stratégie de solution construite au moment de la conception.[9]

c. Le troisième type implique un élément plus complexe, le constructeur de solutions, qui est capable de construire une stratégie de solution complètement nouvelle en utilisant un référentiel de fonctionnalités établies. [9]

d. Enfin, le quatrième type nécessite le moteur d'évolution qui est capable de mettre en place un constructeur de solutions. [9]

7 Conception de système adaptatif

Le problème de la conception de systèmes adaptatifs n'est pas nouveau. Depuis longtemps, de nombreux travaux de recherche ont été menés et diverses approches ont été adoptées pour traiter les propriétés adaptatives recherchées. Cette adaptation est vue comme la capacité de ce dernier à s'adapter à des conditions nouvelles ou différentes. [11]

Elle correspond au processus de modification du système, nécessaire à son bon fonctionnement dans un environnement donné. Une terminologie appropriée signifie que le système est exactement comme prévu dans un environnement spécifique. Par conséquent, nous notons qu'il existe des méthodes basées sur la capacité du système à s'observer, donc répondre aux questions sur son état. Cet attribut est considéré comme essentiel s'adapter aux systèmes complexes. [11]

Avant que les conditions ne changent, on distingue deux situations différentes, la première est la situation où le changement a été planifié, puis une phase de refactorisation du système peut répondre au besoin d'adaptation. Le deuxième cas ne prévoit pas de changement de conditions, alors le système doit modifier une partie ou même la totalité. [11]

8 Les éléments d'un système qui peuvent porter une adaptation

Nous pouvons considérer un système comme une architecture constituée d'un ensemble d'éléments interdépendants. L'adaptation des éléments peut aller du simple paramétrage à une métamorphose complète. Nous spécifions une partie d'un système par le biais d'un élément, qui peut être une fonction, une classe, un composant ou toute autre entité logicielle. Pour les liens, l'adaptation comprend la modification des propriétés du lien ou son application à d'autres paires d'éléments. Ainsi, une adaptation peut être liée à un élément, à une connexion entre deux éléments ou à une architecture entière.[5]

Au niveau architectural, l'adaptation comprend l'ajout, la suppression ou le remplacement d'éléments ou de liens ; comme il peut s'agir d'un redéploiement, qui consiste à modifier la

répartition du système sur un environnement d'exécution composé d'un ou plusieurs sites de déploiement.[5]

9 Boucle de contrôle

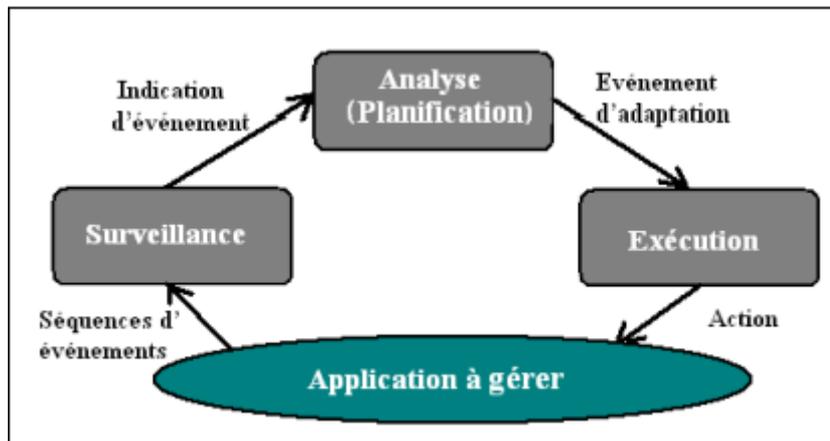


Figure 2 : Boucle de contrôle.[10]

La boucle de contrôle, qui décrit le processus d'ajustement des applications autonomes, est le composant réutilisable du modèle architectural. Cette boucle de contrôle est constituée de trois parties : la surveillance, l'analyse (qui comprend la planification) et l'exécution[10] (voir figure ci-dessus).

1. Le composant **Surveillance** cherche à détecter des séquences d'événements ou des violations spécifiques afin d'invoquer le composant Analyse par le biais d'une indication d'événement.[10]
2. Le composant **Analyse**, quant à lui, analyse les séquences d'indications d'événements pour identifier l'action appropriée. Si une adaptation est nécessaire, ce composant transmet un événement d'adaptation au composant d'exécution.[10]
3. Ensuite, le composant **Exécution** prend les mesures nécessaires en réponse à chaque type d'événement d'adaptation.

10 Méthodologies existantes

Des ingénieurs et des chercheurs de différentes disciplines ont déployé des efforts considérables pour réaliser des systèmes adaptatifs. Cette section aborde diverses méthodes développées au fil du temps. Chaque approche s'inspire d'une discipline spécifique, mettant ainsi en évidence des aspects complémentaires de la construction de systèmes adaptatifs.[7]

10.1 Technologie de contrôle

L'ingénierie de contrôle est une discipline axée sur la conception de systèmes fonctionne comme prévu avec le contrôleur système. Traditionnellement, l'ingénierie de contrôle se concentre sur les systèmes régis par les lois de la physique, tels que l'installation de contrôle physique. Ces dernières années, l'application de la théorie du contrôle il a été étudié dans divers travaux en environnement informatique. De plus, les similitudes entre les deux installations de contrôle physique et les systèmes adaptatifs sont évidents. Les plantes physiques réagissent continuellement à leur environnement pour atteindre des objectifs spécifiques, tout comme les systèmes adaptatifs. Dans cette approche, il existe deux concepts de base, le système cible et le contrôleur. Le contrôleur met en œuvre une stratégie de contrôle qui dicte les signaux de contrôle corrects pour maintenir la sortie du système cible suffisamment proche de la cible souhaitée. Le signal de commande est généralement basé sur la différence entre la sortie du système cible précédent et la cible du système. Le système cible est un modèle analytique basé sur des relations mathématiques qui relient les effets des signaux de commande sur le comportement du système.[11]

Les systèmes basés sur le contrôle reposent sur des boucles de contrôle pour intégrer système cible et les interférences externes. Une technique importante pour organiser les boucles de contrôle dans les systèmes adaptatifs est la boucle MAPE-K, appelée boucle adaptative. En fait, de nombreux travaux ont souligné l'importance et l'application des boucles MAPE-K dans les systèmes adaptatifs basés sur le contrôle. [11]

10.2 Programmation logicielle

Dans cette approche, un langage de programmation à usage général est utilisé pour implémenter système adaptatif. L'une des principales technologies s'appelle des idées qui ont été introduites dans la communauté des langages de programmation, visant à augmenter la flexibilité de programmation et permet le développement de systèmes logiciels fermés qui ne nécessitent pas d'interférence externe. Un système logiciel réflexif est un système qui peut inspecter et modifier son comportement et sa structure. Les langages de programmation prenant en charge la réflexion fournissent de nombreuses fonctionnalités au moment de l'exécution qui facilitent la réflexion, telles que la création de nouveaux types de classes au moment de l'exécution et l'instanciation d'objets à partir de classes qui ne sont pas définies au moment de la compilation. De nombreux langages de programmation à usage général ont déjà Pensez comme JAVA et C#. L'adaptabilité des logiciels est une caractéristique système adaptatif inhérent.[11]

10.3 Intelligence artificielle :

L'intelligence artificielle fournit aux systèmes la capacité d'apprendre, d'améliorer et de prendre des décisions pour effectuer des tâches complexes. Le domaine de l'intelligence artificielle est très vaste, couvrant le traitement du langage naturel, les systèmes multi-agents, l'apprentissage automatique, la théorie de l'utilité, etc. Les systèmes adaptatifs ont quelque chose en commun avec l'intelligence artificielle, à savoir la gestion de scénarios inattendus. Se référant à la boucle adaptative, qui est une exigence essentielle d'un système adaptatif, intelligents les humains peuvent trouver utile dans les deux principaux éléments du cycle d'adaptation, à savoir en savoir plus sur les éléments d'analyse et de planification. La technologie de l'IA peut jouer un rôle central en traitant de grandes quantités de données et en analysant et adaptant prendre une décision. Les techniques d'apprentissage de l'intelligence artificielle peuvent être utilisées pour mieux évaluer et trouver des modèles dans les données observées de l'environnement. En outre, elles peuvent être utilisées pour mieux juger du plan d'adaptation à exécuter en tirant des enseignements des expériences antérieures. [11]

Plusieurs algorithmes d'apprentissage d'intelligence artificielle, tels que l'apprentissage par renforcement et les algorithmes génétiques, peuvent être mis en œuvre dans la phase de planification d'un système auto-adaptatif via le moteur d'adaptation. Le moteur d'adaptation d'un système auto-adaptatif exécute l'étape de planification. [11]

11 Les défis des systèmes auto-adaptatifs

Les systèmes auto-adaptatifs posent de nouveaux défis au développement et à la conception de systèmes logiciels. Cette section vise à identifier les différents défis auxquels les ingénieurs logiciels sont confrontés lors de la mise en œuvre de systèmes auto-adaptatifs.[7]

11.1 Défis de confiance

L'un des principaux défis rencontrés par l'industrie après la sortie des systèmes adaptatifs est le manque de confiance des utilisateurs/administrateurs dans ces systèmes. Ceci est principalement dû à trois problèmes. Premièrement, l'auto-dépendance du système le rend introuvable, de sorte que l'utilisateur est déçu de ce que le système choisit/fait. Une façon de résoudre ce problème est de rapporter les activités et les décisions prises par le système adaptatif à l'administrateur..[11]

11.2 Défis d'analyse

Compte tenu des informations de surveillance, l'objectif principal de l'analyse est de déterminer quand un système est en mauvais état. Un mauvais état fait généralement référence à un mauvais

comportement du système qui nécessite une adaptation. Dans quelle mesure détecte-t-il les conditions défavorables ? Est-il possible de détecter tôt et de prendre les mesures appropriées ? Ce sont de nombreux problèmes qui doivent être résolus. Les tâches analytiques à ce jour ont été considérées comme un défi majeur. En fait, sa complexité a conduit les chercheurs à s'appuyer sur des techniques d'analyse ad hoc et basées sur des règles. Une approche prometteuse consiste à utiliser l'intelligence artificielle et les techniques d'exploration de données pour adopter l'analyse en ligne.[12]

11.3 Défis de planification

Le planificateur prend des captures d'écran de l'état actuel du système et des objectifs du système pour décider d'un plan d'adaptation qui satisfait les contraintes et les objectifs du système. Un plan d'adaptation est une série d'actions qui doivent amener le système d'un mauvais état à un état normal. Malheureusement, cette tâche est difficile sur le plan informatique, de sorte que la plupart des chercheurs s'appuient sur la planification hors ligne. Dans la planification hors ligne, un ensemble de plans est créé au moment de la conception et peut être visualisé via un processus de génération ou de vérification pour répondre aux contraintes du système. Cependant, le véritable défi de la recherche réside dans la planification en ligne, où de nouveaux plans sont synthétisés à la volée à mesure que les objectifs du système changent. D'autres défis incluent la planification d'objectifs multiples et la résolution de conflits, la prise en compte d'informations système incomplètes dans des systèmes décentralisés et la garantie que les comportements transitoires planifiés sont sûrs.[12]

11.4 Défis d'exécution

À ce stade, le système géré utilise l'actionneur pour exécuter le plan d'adaptation. Les principaux points à considérer à cette étape sont : comment faire face à l'échec de l'exécution du plan d'adaptation et à l'interférence entre l'exécution de plusieurs plans d'adaptation. De plus, une étape importante consiste à vérifier que l'exécution du plan d'adaptation est en fait correcte et aboutit au comportement souhaité. La plupart des méthodes existantes reposent sur des exemples limités pour démontrer l'efficacité de leurs méthodes, mais la validation est une étape essentielle. Le comportement adaptatif de ces systèmes détermine le besoin de vérification statique et renforce le besoin de vérification d'exécution. La vérification d'exécution qui s'appuie sur des systèmes adaptatifs est complexe en raison des nombreuses alternatives et chemins d'exécution inhérents à la nature des systèmes adaptatifs. Une combinaison de vérification hors ligne et en ligne semble être une solution faisable mais difficile.[12]

12 Systèmes existants représentatifs

Plus réactif aux changements d'exécution en permanence, le développement de systèmes présentant des propriétés adaptatives devient exigeant.[5]

L'adaptation repose essentiellement sur le fait qu'elle est constituée d'entités séparables, c'est-à-dire qu'elle doit être décomposable. Elle repose également sur une autre propriété très importante, la capacité du système à s'observer et donc à répondre aux changements de son état.[5]

a) Projet SMARTHERM

SMARTHERM est un projet R&D exclusif de SYL-VERSOFT Inc. pour développer des systèmes de contrôle par un ordinateur de température ambiante basé sur la technologie de l'intelligence artificielle. Né au début des années 90, le projet opère dans des laboratoires et des maisons modèles depuis plus de 8 ans. Son processus de développement a été relativement long et laborieux, car il a dû être testé dans des environnements d'utilisation réels, en particulier dans les conditions hivernales difficiles qui prévalent au nord du parallèle à 49 degrés.[13]

Par temps froid, plusieurs phénomènes très complexes se conjuguent pour faire perdre l'accumulation de chaleur dans une habitation, et le fait qu'il est difficile pour tout système de chauffage d'atteindre et de maintenir un certain niveau de chaleur dans ce logement. Les phénomènes les plus déterminants sont :[13]

– Température extérieure (perte de chaleur exponentielle lorsque la température extérieure diminue).[13]

– La vitesse et la direction du vent (le facteur vent détermine le corps La température effective est inférieure de quelques degrés à la température nominale).[13]

– le degré ou le manque d'ensoleillement (le soleil provoque une augmentation importante de la température du corps exposé au soleil, surtout s'ils sont dans une position scellée entreprise).[13]

– La température intérieure souhaitée (la difficulté d'atteindre et de maintenir la température ambiante à l'intérieur de la maison augmente de façon exponentielle avec chaque degré d'augmentation, voir Figure au-dessous).[13]

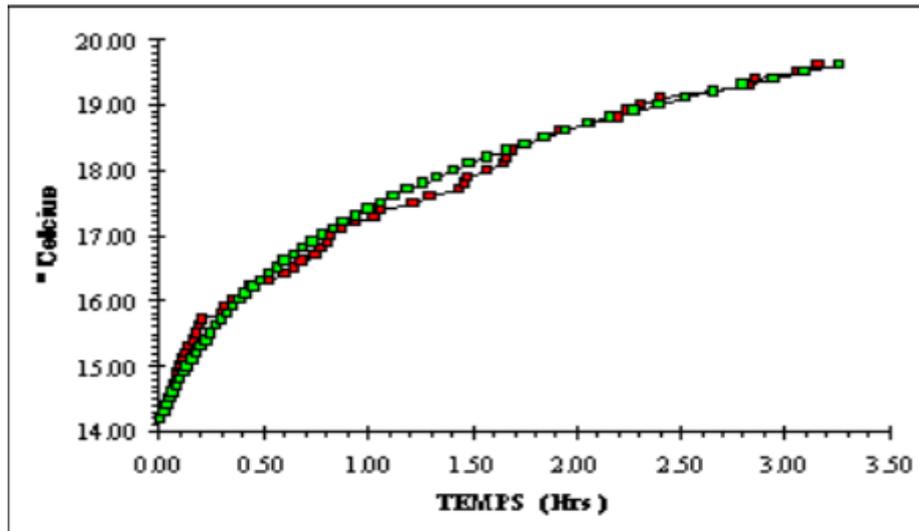


Figure 3: Courbe de difficulté du maintien de la température ambiante.[13]

– Facteur d'isolation de la maison (état de l'isolation, surface des fenêtres, orientation de la maison, etc.).[13]

– Habitudes des occupants (seuil de tolérance au froid, fréquence d'entrée et de sortie, ouverture/fermeture des fenêtres, ouverture/fermeture des rideaux ou des stores, etc.).[13]

SMARTHERM est un système adaptatif qui, grâce à divers capteurs et à l'intelligence artificielle, s'efforce de prendre en compte la plupart des facteurs environnementaux susceptibles d'affecter l'obtention et le maintien du confort de travail calorimétrique souhaité.[13]

La température requise dans une habitation par temps froid. Dans sa version de base, il dispose d'un capteur de température interne et d'un capteur de température externe. Dans sa version la plus avancée (SMARTHERM Plus) il a en plus des capteurs de vent (anémomètres) et des capteurs d'ensoleillement (capteurs solaires).[13]

b) Projet Lumière

L'objectif du projet Lumière est d'aider les utilisateurs de logiciels. Il s'agit d'un projet de recherche lié au « compagnon » utilisé par Microsoft.[13]

Le projet est basé sur un modèle bayésien qui gère différentes opérations et demandes.[13] le projet de recherche comprend :

1. Construction d'un modèle bayésien basé sur des observations de différents comportements et demandes d'utilisateurs.[13]

CHAPITRE 1 : Généralités sur l'auto-adaptation des systèmes logiciel

2. Améliorer la vitesse d'accès aux différents événements utilisateur.[13]

3. Développer un langage pour convertir les événements en variables observées utilisées dans les modèles bayésiens.[13]

4. Développement des profils selon l'expertise de l'utilisateur.[13]

5. Développement de l'architecture de l'interface utilisateur intelligente.[13]

L'objectif est de s'adapter automatiquement aux utilisateurs au fil du temps en fonction de leurs besoins et de leurs activités.

c) Projet Astrolab :

Le projet IBM Astrolab permet de restaurer un état différent systèmes hétérogènes qui existent sur un réseau local pour effectuer dynamiquement des mises à jour de configuration. Le système est évolutif et fonctionne d'une certaine manière décentralisée. Les mises à jour de configuration impliquent des pilotes, des logiciels et paramètres spécifiques à la gestion LAN. Le projet permet l'abstraction de la gestion de systèmes hétérogènes en s'adaptant à l'évolution des utilisateurs, des logiciels et du matériel.[13] la figure ci-dessous représente un exemple d'un Astrolab.



Figure 4 : Exemple représentatif d'un Astrolab.[14]

13 Bilan de ces recherches

Les différents projets présentés dans cette section montrent que les systèmes existants évoluent vers une adaptation au service des utilisateurs. Un dénominateur commun à ces projets est que le système évolue dans un environnement instable, mais reste borné. Une autre chose que ces projets ont en commun est que le cœur du système ne se modifie pas en temps réel pour s'adapter, une

CHAPITRE 1 : Généralités sur l'auto-adaptation des systèmes logiciel

configuration de base et fixe est nécessaire pour un bon fonctionnement logiciel. L'architecture de ces projets est très spécifique, le système n'est pas non générique et utilisable pour de nouveaux problèmes.[13]

La planification dynamique des processus permettra l'auto-modification interne, résultant en des actions qui s'adaptent aux différentes situations possibles dans l'environnement. Cela permet également de changer la configuration de base sans demander à l'utilisateur des actions spécifiques, assurant ainsi la généralité du système.[13]

14 Conclusion

Dans ce chapitre, nous avons introduit les concepts, les propriétés et les défis dans le domaine des systèmes auto-adaptatifs de manière simple, La compréhension de cette nouvelle technologie contribuera sans aucun doute à l'élaboration de notre approche, Dans le chapitre suivant nous parlerons sur E-Learning, et quelques plateforme E-Learning ainsi que les plateformes E-Learning adaptatifs.

CHAPITRE 2 :
Etat de l'art sur E-Learning

1 Introduction

E-Learning ou « L'apprentissage en ligne » ou « L'apprentissage électronique » est devenu l'une des bases dont le monde a besoin maintenant, est c'est l'un des sujets intéressent pour lesquels un avenir radieux est attendu.

Début 2019, le virus dit corona a fait surface pour la première fois, ce qui a nécessité la mise en quarantaine d'un monde entier, y compris des étudiants et des travailleurs, et l'apprentissage à distance est devenu une nécessité indispensable.

Dans ce chapitre, nous approfondissons un peu le concept E-Learning, leur évolution, leurs avantages et les inconvénients et quelques plateformes E-Learning.

2 Evolution de E-learning

L'enseignement à distance ou l'apprentissage à distance comprend la technologie et donne l'accès aux programmes éducatifs pour les étudiants suivants séparés par le temps et l'espace.[15]

Il existe plusieurs manières de dispenser une formation à distance : Lettres papier, cassettes vidéo éducatives, enseignement de l'informatique (enseignement multimédia, utilisation d'Internet pour l'éducation en ligne), etc.[15]

L'apprentissage électronique n'est utilisé que depuis 1999. cependant, les concepts de l'apprentissage électronique ont été soigneusement répertoriés tout au long de l'histoire, des preuves suggérant que les premiers types d'apprentissage électronique existaient déjà au XIXe siècle.[7]

Avec l'avènement de l'ordinateur et d'Internet à la fin du vingtième siècle, les technologies et les techniques de diffusion de l'apprentissage électronique ont gagné en popularité.[7]

Puis, au cours de la décennie suivante, alors que les gens accédaient à une variété de connaissances et de possibilités d'apprentissage en ligne, les environnements d'apprentissage virtuel ont commencé à s'épanouir.[7]

Les entreprises ont commencé à adopter l'apprentissage en ligne pour former leur personnel dans les années 2000.[7]

Les travailleurs, qu'ils soient nouveaux ou expérimentés, peuvent désormais accroître leurs connaissances du secteur et étendre leurs capacités. Les particuliers peuvent accéder depuis chez eux à des programmes qui leur permettent d'acquérir un diplôme en ligne et d'améliorer leur vie en améliorant leurs connaissances.[7]

3 Définition « E-Learning »

[16] ont analysé plus de 50 articles sur le sujet du e-Learning, il nous a fait le résumé suivant :

	(A) Apprentissage autonome <i>Computer-Based Instruction/ Learning / Training (CBI/L/T)</i>	(B) Apprentissage collaboratif / Apprentissage coopératif <i>Computer-Mediated Communication(CMC)</i>
(1) Apprentissage en ligne Communication synchrone en temps réel	Surf sur Internet, accès à des sites contenant des informations utiles pour une formation (connaissances ou expériences) en ligne (exerciceur web)	Chat rooms avec ou sans vidéo (IRC; tableau blanc, WEB TV), Audio/vidéoconférence (streaming audio et vidéo)
(2) Apprentissage hors ligne Communication asynchrone	Téléchargement d'objet pour une utilisation locale <i>LOD (Learning Object Download)</i>	Communication asynchrone par courriel, liste de discussion ou forum via des plates-formes

Figure 5 : Structure du e-Learning.[16]

D'après[16]l'apprentissage en ligne peut prendre de nombreuses formes, basées sur une communication de manière synchrone ou asynchrone et apprendre selon des modes autonomes ou collaboratifs, Apporte une richesse de complexité au concept.

Selon[17], le terme e-learning a plusieurs définitions, mais aucune est ne consiste. Pour délimiter ses limites, nous allons examiner de plus près deux concepts clés du l'abréviation "e" qui signifie le mot "électronique" ou "en ligne" et "Learning" traduits en français "Apprentissage", le résultat est apparemment "e-learning" ou "apprentissage électronique".

Une sorte d'incompréhension vient du fait que [17]avaient trouvé plusieurs noms pour chaque partie du concept e-learning, comme le montre la figure ci-dessous, ils peuvent facilement obtenir plus d'une vingtaine de combinaisons telles que : formation virtuelle, e-learning, etc.

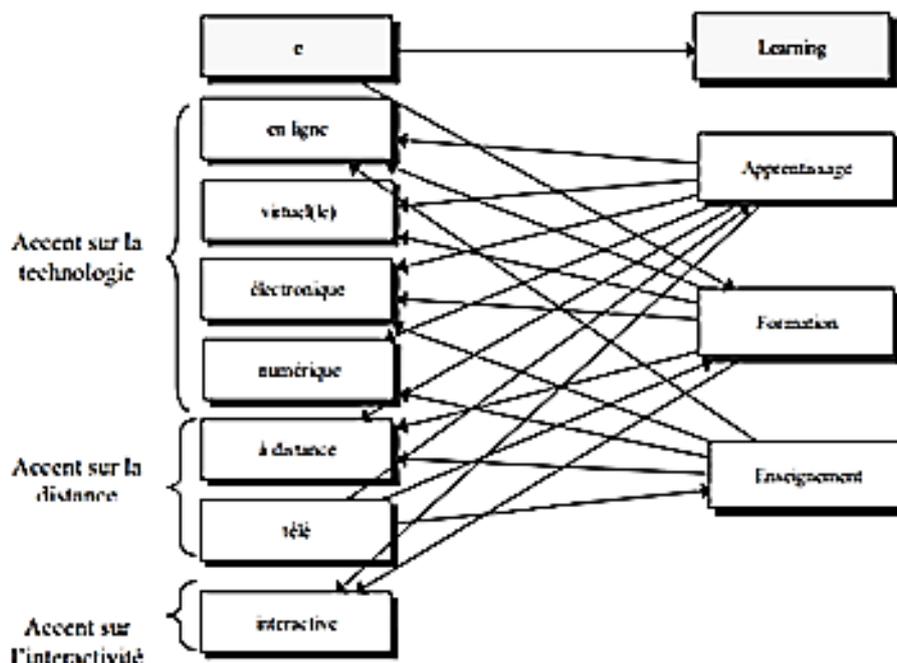


Figure 6: quelques termes souvent associés au E -Learning. [17]

Dans un premier temps, ils abordèrent la partie « Learning », et constatent que la définition de ce concept utilise souvent trois termes : le premier est « l'apprentissage » et le second est « formation » traduit en anglais « training » Enfin, c'est « l'éducation ». Le choix n'est généralement pas évident, mais pour la plupart des termes « Apprendre » fait référence au concept d'éducation à travers une pédagogie dont « l'apprenant » est le noyau principal. [17]

Dans un second temps, ils continuent à analyser le segment « e » reflète les 3 termes. Le premier cas signifie "électronique" ou "en ligne" ou même "virtuel". Tous ces concepts expliquent que l'apprentissage se fait à travers des outils technologiques. [17]

Considère le e-Learning comme un apprentissage utilisant la technologie. Le deuxième cas, faisant référence à la distance "temporelle et physique" entre l'apprenant et le formateur : généralement exprimé en termes de formation à distance, il faut cependant distinguer e-Learning et formation à distance : la formation à distance peut être réalisée par imprimés, documents audio-visuels, etc., donc via des technologies non numériques, alors que l'apprentissage en ligne repose par nature sur des outils technologiques basés sur le Web. Enfin, il y a le dernier cas de "l'apprentissage interactif", qui met l'accent sur la capacité de l'apprenant à interagir avec les outils informatiques. [17]

CHAPITRE 2 : Etat de l'art sur E-Learning

Cette perspective terminologique permet de donner une définition assez complète du e-learning, qui peut s'énoncer comme suit : le e-learning est la capacité d'apprendre à travers l'utilisation des technologies de réseau.[17]

4 Les avantages et les inconvénients

L'utilisation des dispositifs d'apprentissage en ligne est de plus en plus courante, ce qui a entraîné l'émergence de nombreux refrains d'utilisation.[18]

Pour énumérer les nombreux avantages et inconvénients du e-learning, il faut considérer trois acteurs : l'apprenant, l'entreprise et le formateur.[18]

4.1 Les avantages

a) Permet la formation « juste à temps » :

La formation virtuelle permet aux apprenants d'étudier et de rafraîchir leurs connaissances à leur guise, plutôt que d'attendre qu'un cours spécifique soit dispensé à un moment précis.[15]

b) Offre une formation flexible :

Les modes de prestation de l'apprentissage en ligne offrent une variété d'options pour recevoir une formation, comme dans une salle de classe traditionnelle aménagée à cette fin, dans un bureau réservé spécifiquement à l'apprentissage individuel, ou à des postes de travail installés dans l'entreprise pour les employés qui n'ont pas accès à un ordinateur, et même à la maison.[15]

c) Aide à réduire les coûts de formation :

Bien que l'apprentissage virtuel exige un investissement initial important, son utilisation permet souvent de réaliser d'importantes économies. Ces économies sont principalement réalisées grâce à la réduction des frais de déplacement des apprenants, à la diminution des honoraires des formateurs, à la réduction du temps consacré à l'apprentissage et à la réduction de la productivité en rendant la formation disponible sur le lieu de travail.[15]

d) Uniformise et personnalise l'apprentissage :

L'apprentissage en ligne permet d'uniformiser la formation tout en mettant des cours et des sections de cours à la disposition des employés en fonction de leurs besoins spécifiques.[15]

e) Permet la diffusion simultanée de l'apprentissage à un large public :

Si quelqu'un introduit un nouveau produit ou service et qu'il doit enseigner à un grand nombre de personnes en même temps, l'apprentissage en ligne est la méthode à utiliser. Il peut atteindre à tout moment toute personne ayant accès à un ordinateur relié à l'Internet ou à leur intranet. Il peut ainsi assurer que tout le monde reçoit la même instruction au même moment.[15]

4.2 Les inconvénients

a) La diffusion de cours en ligne nécessite l'utilisation d'équipements multimédias :

L'équipement de l'entreprise doit être capable de distribuer les informations du cours. Cela concerne particulièrement les postes de travail des utilisateurs (PC récents, logiciels compatibles installés, éventuellement un réseau avec une bande passante suffisante). Compte tenu de l'évolution des technologies, ce déficit tend à s'estomper avec le temps.[15]

b) La mise en place de l'infrastructure technologique et la création du contenu sont coûteuses :

Pour mettre en place une politique d'e-learning, un investissement financier est nécessaire. Cet investissement à coût fixe et à risque élevé remplace le paiement de services de formation (coût variable et faible risque). De plus, le marché n'est pas encore consolidé et l'offre de cours spécialisés varie selon les domaines.[15]

c) Un accès informatique est nécessaire :

L'utilisation d'outils informatiques limite la diffusion de l'e-learning à un sous-ensemble d'employés. C'est un obstacle dans le secteur industriel, par exemple.[15]

d) L'apprentissage en ligne réduit les interactions interpersonnelles :

Certains systèmes de communication, comme le langage corporel, ne peuvent être reproduits, malgré leur importance dans la diffusion des connaissances.[15]

5 Quelques plateformes E-Learning

Le nombre de plateformes d'apprentissage en ligne augmente de jour en jour et il est presque impossible de décrire une liste complète de ces systèmes. La partie suivante comprend des plateformes d'apprentissage en ligne parmi les plus couramment utilisés :

Moodle¹ :

Est un système de gestion de cours open source, ses origines remontent années 1990. En 2003, a été fondé Lancé pour fournir un soutien aux ventes, la gestion Hébergement, conseil et autres services. Depuis 2005, une équipe de développement employé par Moodle, et une grande communauté de développeurs et d'organisations soutenir les contributions aux projets de code source, idées, etc.[19]

Comprend des fonctionnalités d'administration personnalisables pour créer des sites Web privés avec des cours en ligne pour les éducateurs et les formateurs afin d'atteindre leurs objectifs d'apprentissage. Moodle permet d'étendre et de personnaliser l'environnement d'apprentissage à l'aide de plugins communautaires.[7](Voir Figure 7)



Figure 7 : une simple interface de Moodle.

ATutor² :

Est un système d'apprentissage à distance open source qui prend en charge la gestion de contenu, en particulier les problèmes d'accessibilité et d'adaptabilité. Il a été publié en 2002

¹ Accessible sur : <https://moodle.org>

² Accessible sur : <http://atutor.ca/>

CHAPITRE 2 : Etat de l'art sur E-Learning

après deux études visant à évaluer l'accessibilité pour les utilisateurs ayant des troubles d'apprentissage. (Voir Figure 8)



Figure 8: une simple interface d'ATutor.

Dokeos³ :

Est une suite complète d'apprentissage à distance. Il permet de produire des cours, d'organiser des formations, d'assurer l'interaction et le suivi des apprenants. La plupart des parties de la plate-forme sont téléchargeables gratuitement, tandis que d'autres sont disponibles dans le commerce. (Voir Figure 9)

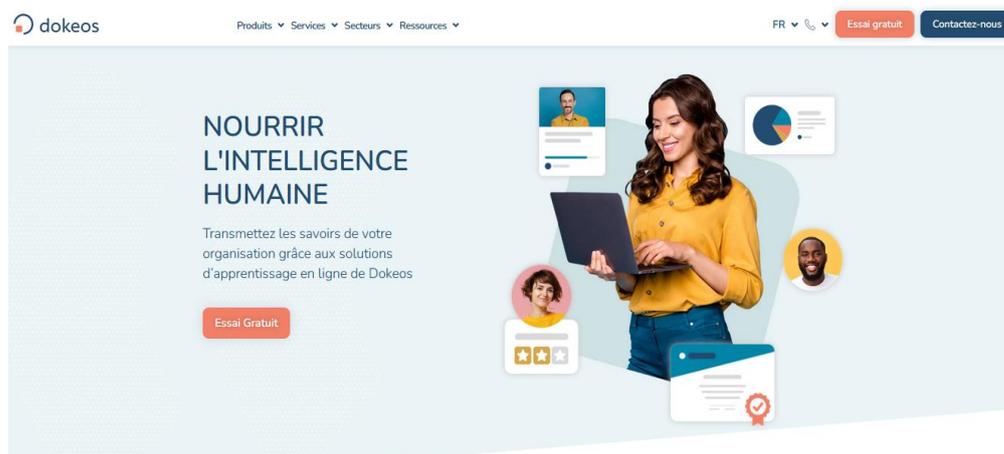


Figure 9: une simple interface de Dokeos.

³ Accessible sur : <http://www.dokeos.com/>

Coursera⁴ :

Il s'agit d'une plateforme d'apprentissage en ligne fondée par les professeurs d'informatique de l'Université de Stanford Andrew Ng et Daphne Kohler en 2012, proposant un grand nombre de cours ouverts en ligne, de majors, de diplômes, de cours professionnels et de masters.[7]

Coursera s'associe à des universités et à d'autres organisations pour proposer des cours, des certifications et des diplômes en ligne dans diverses disciplines telles que l'ingénierie, la science des données, l'apprentissage automatique, les mathématiques, les affaires, la finance, l'informatique, le marketing numérique, les sciences humaines, la médecine, la biologie, les sciences sociales, etc. plus de 3000 cours, offrant aux étudiants un large éventail d'informations et d'expériences dans différents domaines.[7](Voir Figure 10)

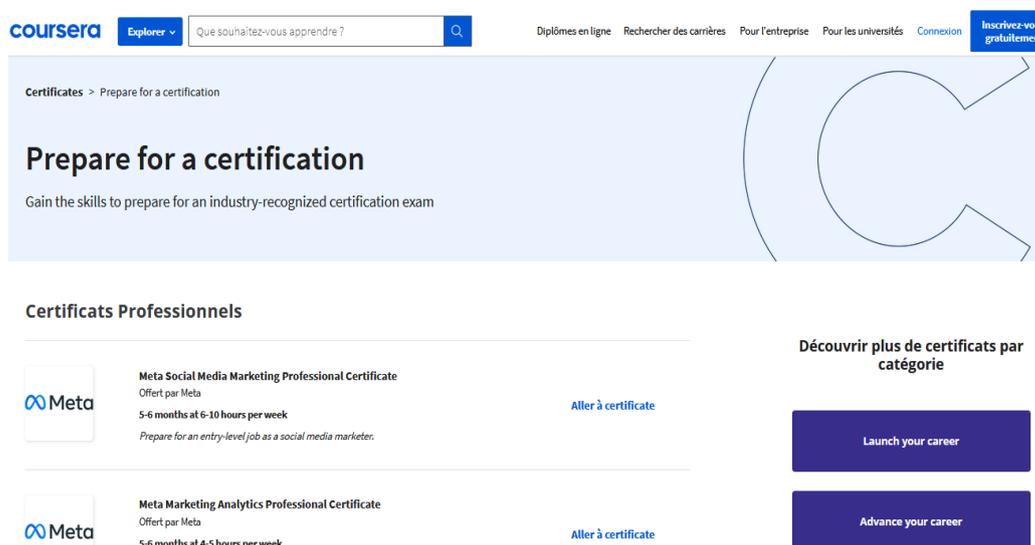


Figure 10: une simple interface de Coursera.

6 L'adaptation dans le E-Learning

Avec la croissance des plateformes et des applications éducatives, le terme d'adaptation a évolué. Il s'agit d'une caractéristique plus cruciale pour l'e-learning avancé. Il est difficile d'obtenir l'aide avisée et individualisée qu'un enseignant ou un camarade de classe peut apporter dans une salle de classe typique.[18]

En outre, étant donné que le support pédagogique basé sur le Web sera utilisé par un éventail d'élèves bien plus large que n'importe quel outil éducatif "autonome", il est essentiel qu'il soit

⁴Accessible sur : <http://www.coursera.org>

CHAPITRE 2 : Etat de l'art sur E-Learning

adaptable. Une ressource pédagogique préparée en pensant à un type spécifique d'utilisateurs peut ne pas convenir à d'autres utilisateurs.[18]

Dans cette partie on va présenter les différents cas d'adaptation dans le E-Learning. Qui peuvent s'effectuer sur trois niveaux :

6.1 L'adaptation de la navigation :

Cela comprend la modification de l'ordre dans lequel les liens apparaissent, les annotations des liens, la cible des liens et même le nombre de liens présentés à l'apprenant. Cela aide à prendre en charge la navigation, empêchant les apprenants de suivre des chemins de navigation sans rapport avec leurs tâches ou leurs objectifs.[19] Dans ce sens on a les techniques suivant :

- **Le guidage direct :** Le but de cette technique est d'identifier la destination de l'apprenant. Il est basé sur la stratégie "la prochaine est la meilleure" (utilisez le bouton "suivant" pour vous guider) ;[19]
- **Annotation du lien :** le lien sera annoté avec du texte, une couleur ou une icône ;[19]
- **Le masquage des liens :** selon le modèle pour chaque apprenant, le système masque le lien hypertexte dont la destination est jugée ne pas être adapté à l'apprenant ;[19]
- **Ordre des liens :** il s'agit de changer l'ordre d'apparence des liens est affichée de manière à ce qu'ils soient affichés par ordre décroissant de pertinence.[19]

6.2 Adaptation du contenu :

Cela comprend le choix de différentes ressources, c'est-à-dire des images, des textes, des animations, des vidéos, etc. ; dépend du modèle de utilisateurs. Il s'agit par exemple de présenter le même contenu en format vidéo ou texte selon la préférence d'utilisateurs. Parmi les techniques d'adaptation utilisées pour ce mécanisme, on distingue :[19]

- **Le texte extensible :** cette technique peut déterminer quels fragments de contenu développer et réduire selon le modèle utilisateurs ;[19]
- **Les fragments conditionnels :** selon le concept des informations fournies par le modèle d'utilisateurs et la relation du modèle de domaine, la sélection d'une partie des informations à afficher par le système est déterminée ;[19]
- **Les pages variantes :** cette technique comprend gardez le contenu de la page adapté chaque niveau ;[19]

CHAPITRE 2 : Etat de l'art sur E-Learning

Et nous pouvons parler en général et brièvement que tous les utilisateurs (un débutant, un expert) n'ont pas des caractéristiques et des informations communes et pour cela le contenu des pages doit adapter cette différence, donc l'adaptation du contenu consiste à sélectionner l'information qui dépend de l'état courant de utilisateurs de page.

6.3 Adaptation de la présentation :

Il fournit différentes dispositions d'éléments Interface de communication apprenant. Il Par exemple, planifier Différents supports, différentes options de couleur Texte, taille de police ou taille d'image. Cette adaptation appelé aussi Adaptation des IHM.[19].

Nous spécifions deux méthodes pour cette catégorie :

- **Les multi-langues** :l'objectif d'une approche multilingue est l'adaptation de la mise en page la langue préférée de l'apprenant ;[19]
- **La mise en page** : méthode de mise en page Inclure toutes les alternatives possibles nécessaires pour s'adapter à la présentation (couleur, taille, police, etc.).[19]

7 Les plateformes E-Learning adaptatives :

Cette partie s'intéresse à quelques plateformes E-Learning adaptatives :

AHA⁵ (A **Adaptive** **H** **Hypermédia** **A** **rchitecture) :**

C'est un système générique d'hypermédia basé sur l'adaptation des pages. Le moteur qui maintient le modèle de l'apprenant permet de générer du texte et d'ajuster la structure des liens en supprimant, masquant ou annotant les liens.(**Voir figure 11**)[20]

⁵Accessible sur : <https://www.aha.io/>

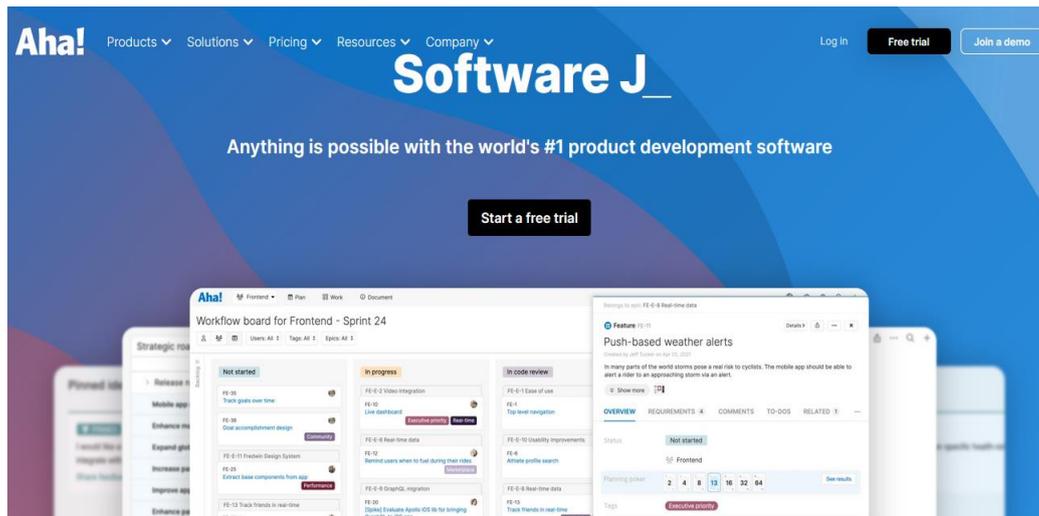


Figure 11: une simple interface d'AHA.

ELM-ART⁶(**E**pisodic **L**earner **M**odel – **A**daptive **R**emote **T**utor):

Il s'agit d'un système hypermédia adaptatif pour l'apprentissage des langages LISP.[21]

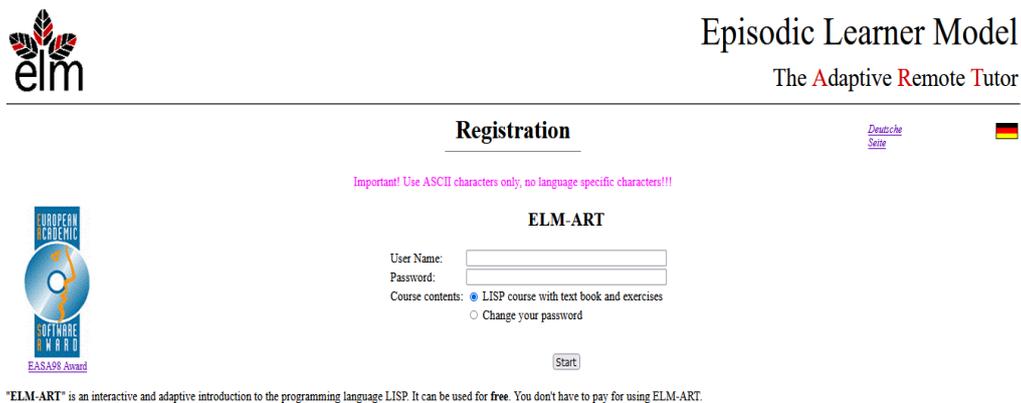


Figure 12: une simple interface d'ELM-ART

Avec un support de navigation adaptatif, ELM-ART se compose d'une série de leçons utilisant principalement un guidage direct et des annotations liées. Des solutions personnalisées et des exemples basés sur les réponses des apprenants sont également fournis (**Voir figure 12**). ELM-ART peut être testé gratuitement en ligne sur le lien suivant : <http://art2.ph-freiburg.de/art/login-e.html> ;[21]

iWeaver :

⁶Accessible sur : <http://art2.ph-freiburg.de/art/login-e.html>

CHAPITRE 2 : Etat de l'art sur E-Learning

Est un projet de doctorat qui vise à créer un environnement d'apprentissage flexible et adaptable pour le langage de programmation Java.[22] (**figure 13**)

Lorsqu'un utilisateur accède initialement au système, celui-ci établit un profil d'apprenant en analysant les styles d'apprentissage au moyen d'une série de questions à choix multiples.[22]

Par la suite, les utilisateurs recevront des recommandations personnalisées et une perspective adaptée des ressources d'apprentissage disponibles.[22]

Les approches de navigation adaptative et de présentation adaptative du contenu sont combinées dans i Weaver.[22]

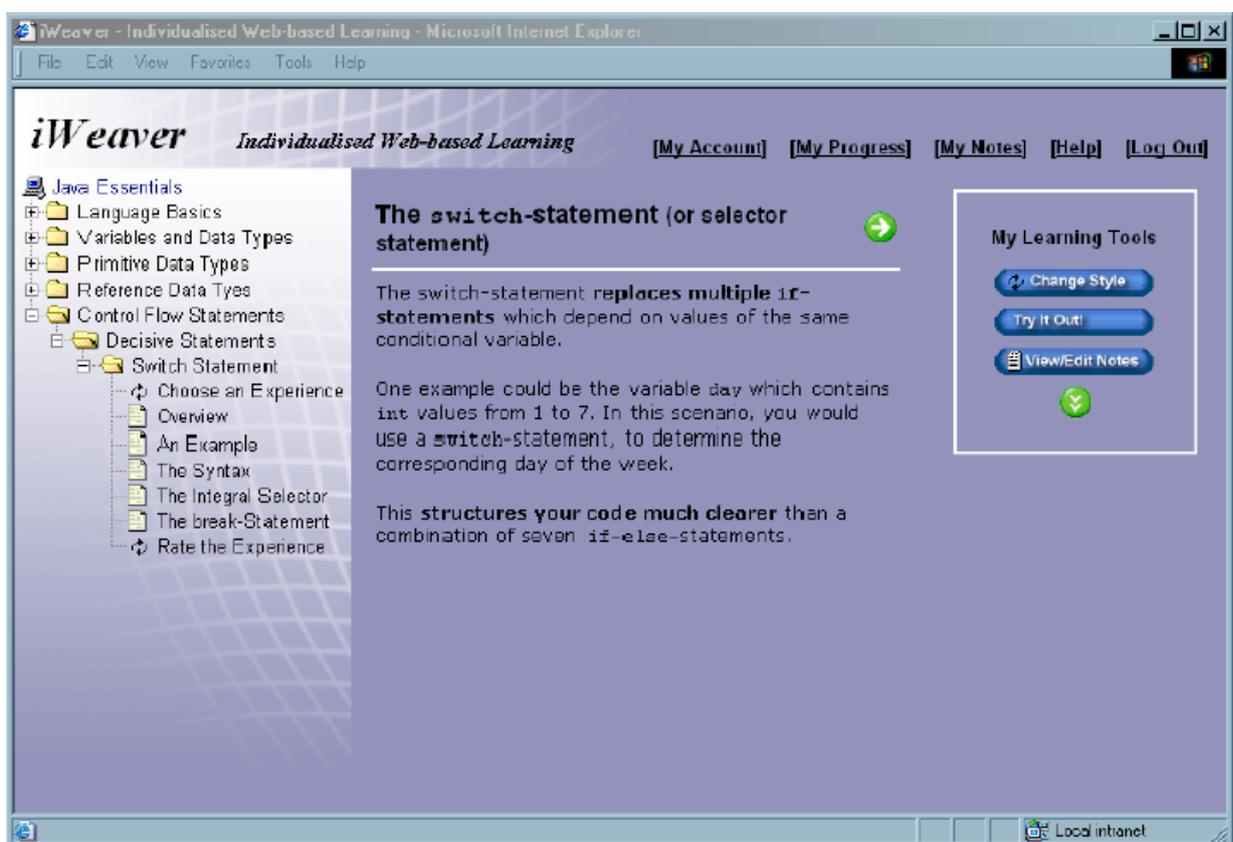


Figure 13:une simple interface d'i Weaver.

Net Coach⁷:

Est une extension d'ELM-ART, avec son propre système d'écriture qui permet la création de cours adaptatifs.[23]

⁷Accessible sur : <https://netcoach.paris/>

CHAPITRE 2 : Etat de l'art sur E-Learning

En général, tout le contenu du cours est structuré en arborescence et peut être consulté librement par l'apprenant.[23]

En outre, la technologie permet de personnaliser le cours par le biais d'un séquençage adaptatif du programme et d'une annotation adaptative des liens.[23](Voir Figure 14)

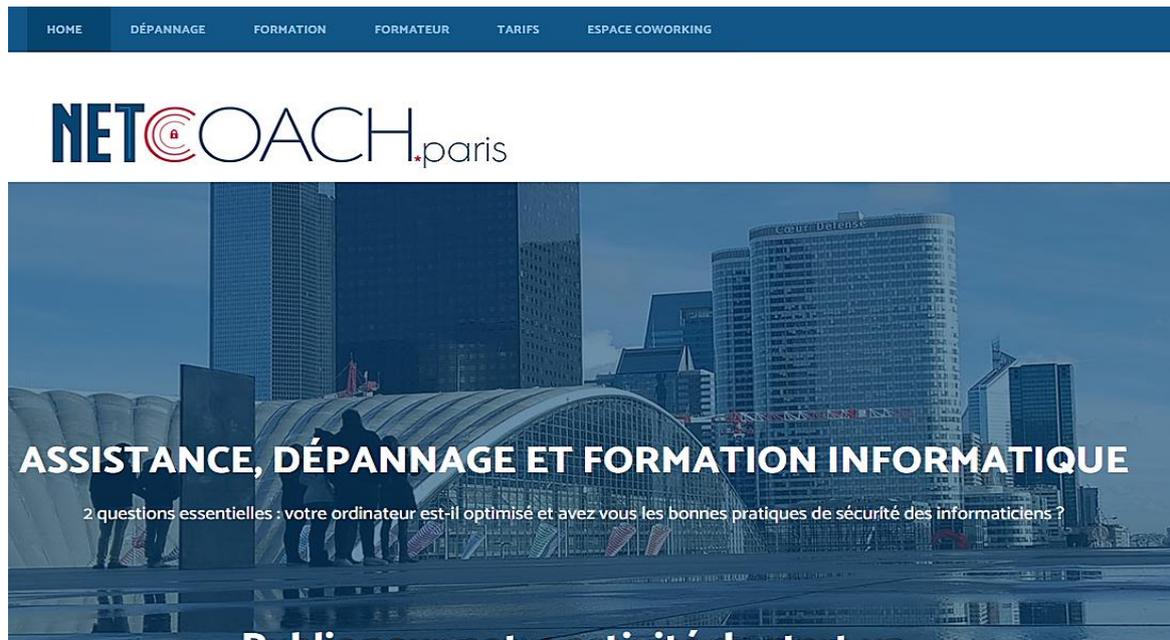


Figure 14:une simple interface de Net Coach.

Après la citation de quelques plateformes e-learning populaires et d'autres adaptatives, le tableau ci-dessous illustre les différentes adaptations effectuées sur ces plateformes adaptatives :

CHAPITRE 2 : Etat de l'art sur E-Learning

Tableau 1 : Les différentes adaptations effectuées sur les plateformes adaptatives.

La Plateforme		AHA	ELM-ART	I Weaver	Net Coach
L'adaptation					
De navigation	Le guidage direct		+	+	+
	Annotation du lien	+	+	+	+
	Le masquage des liens	+		+	
	L'ordre des liens			+	
Du contenu	Le texte extensible			+	+
	Les fragments conditionnels		+	+	+
	Les pages variantes		+	+	+
De présentation	Les multi-langues				
	La mise en page	+			

8 Discussion :

Dans les sections précédentes nous avons étudié les différents types d'adaptation dans le E-Learning, on a l'adaptation de la navigation, l'adaptation du contenu, de la présentation, on a présenté quelques plateformes E-Learning adaptatives (AHA, ELM-ART, i Weaver, Net Coach), et nous avons dit que la plateforme AHA utilise l'adaptation des navigations et présentation, et les autres plateformes basées sur l'adaptation de navigation et du contenu.

CHAPITRE 2 : Etat de l'art sur E-Learning

Dans notre approche, nous avons pris soin sur l'adaptation de présentation notamment l'adaptation des multi-langues dont l'objectif est la mise en page de la langue préférée de l'apprenant qui le convient, et l'adaptation de la mise en page qui inclure toutes les alternatives possibles nécessaires pour s'adapter à la présentation (couleur, taille, police, etc.), nous nous sommes également concentrés sur l'adaptation du contenu notamment l'adaptation de texte extensible qui déterminer quels fragments de contenu développer et réduire selon le modèle utilisateurs .

9 Conclusion :

Dans ce chapitre nous avons présenté quelque définition par rapport aux *E-Learning* et les avantages et les inconvénients offerts par lui, puis nous avons appris quelques plateformes E-Learning et les plateformes E-Learning adaptatives.

CHAPITRE 3 :

**Algorithme d'optimisation par les
essaims particulaires PSO.**

1 Introduction

La recherche d'un état optimal est l'un des principes les plus fondamentaux de notre monde, qui est également significatif dans le monde informatique, l'industrie et d'autres domaines. De nombreux problèmes scientifiques, sociaux, économiques et techniques ont, en fait, des paramètres qui peuvent être modifiés pour obtenir un meilleur résultat. Il s'agit de problèmes d'optimisation, et leur résolution est un sujet important de la recherche opérationnelle, Des techniques ont été conçues pour résoudre ces problèmes ces techniques appelée heuristiques et méta heuristiques, les méthodes exactes.[24]

Ce chapitre est consacré à l'étude des métaheuristiques notamment l'algorithme Optimisation par les essais particulière (PSO) que nous avons utilisé dans notre approche et on va présenterons leur caractéristique et fonctionnement d'une manière détaillée.

2 Problème d'optimisation

En général, un problème d'optimisation consiste en une collection de variables X , une fonction objective F et un ensemble de contraintes d'égalité (ou d'inégalité) que les variables doivent respecter. Un espace de recherche E est formé par une collection de solutions probables à un problème, où chaque dimension correspond à une variable. L'espace de recherche E est limité puisque le décideur définit le domaine exact de chaque variable, notamment pour des considérations de temps de calcul. Selon la requête, la fonction objectif F est soit minimisée, soit maximisée. Les problèmes d'optimisation peuvent être statiques ou dynamiques (la fonction objective varie dans le temps), mono- ou multi-objectifs (plusieurs fonctions objectives doivent être optimisées), et avec ou sans restriction. Par conséquent, les problèmes d'optimisation peuvent être par exemple, à la fois continus et dynamique. [24]

Mathématiquement on cherche $x^* \in E$ [25]

$$f(x^*) = \min (\text{ou max}) f(x) \dots \dots \dots (1)$$

3 Algorithmes de résolutions

Il existe de nombreuses méthodes déterministes (ou exactes) pour résoudre certains types de problèmes d'optimisation et obtenir une solution optimale au problème dans un délai raisonnable. Ces méthodes exigent que la fonction objective présente de nombreuses caractéristiques, telles que la convexité, la continuité ou la dérivabilité. [24]

3.1 Les méthodes exactes

Les méthodes exactes (appelées aussi complètes) produisent une solution optimale pour une instance de problème d'optimisation donné. Cependant, ces méthodes peuvent rapidement devenir coûteuses en temps d'exécution, en particulier pour les problèmes NP-difficiles. En effet, temps de traitement et complexité du problème sont souvent liés (plus il est complexe, plus le temps d'exécution est important).[26]

3.2 Les méthodes approchées

a) Heuristique

Les approches exactes ne sont pas toujours viables pour un problème particulier en raison d'un certain nombre de restrictions, telles que le temps de calcul généralement important ou la difficulté, voire l'impossibilité dans certaines situations, d'une spécification distincte du problème. [24]

Une heuristique est un algorithme qui fournit rapidement (en temps polynomial) des solutions approximatives et pratiques, mais pas nécessairement optimales, à des problèmes d'optimisation complexes. C'est l'inverse d'un algorithme exact, qui fournit la meilleure solution à un problème donné. [24]

b) Métaheuristiques

La métaheuristique est composée de deux mots grecs : méta et heuristique. Le suffixe méta signifie "au-delà", ou "d'un degré supérieur ». Elles sont des procédures qui sont souvent influencées par la nature. Contrairement aux heuristiques, elles peuvent être utilisées pour une variété de problèmes. Par conséquent, nous pouvons affirmer qu'elles sont des heuristiques contemporaines de plus haut niveau dédiées spécifiquement à la résolution de problèmes d'optimisation. Leur objectif est d'atteindre un optimum global tout en évitant les optimums locaux.[27]

Les métaheuristiques regroupent des méthodes qui peuvent se diviser en deux classes :

- **Méta-heuristiques solution unique** : Ces approches recherchent la meilleure réponse en ne traitant qu'une seule solution à la fois. [27]

- **Métaheuristiques à population de solutions** : À chaque itération, ces approches emploient une population de solutions jusqu'à ce que la solution globale soit atteinte. [27]

4 Classification des algorithmes de résolution :

Les méthodes d'optimisation se divisent en deux catégories : les méthodes exactes et les approchées (voir Figure 15).

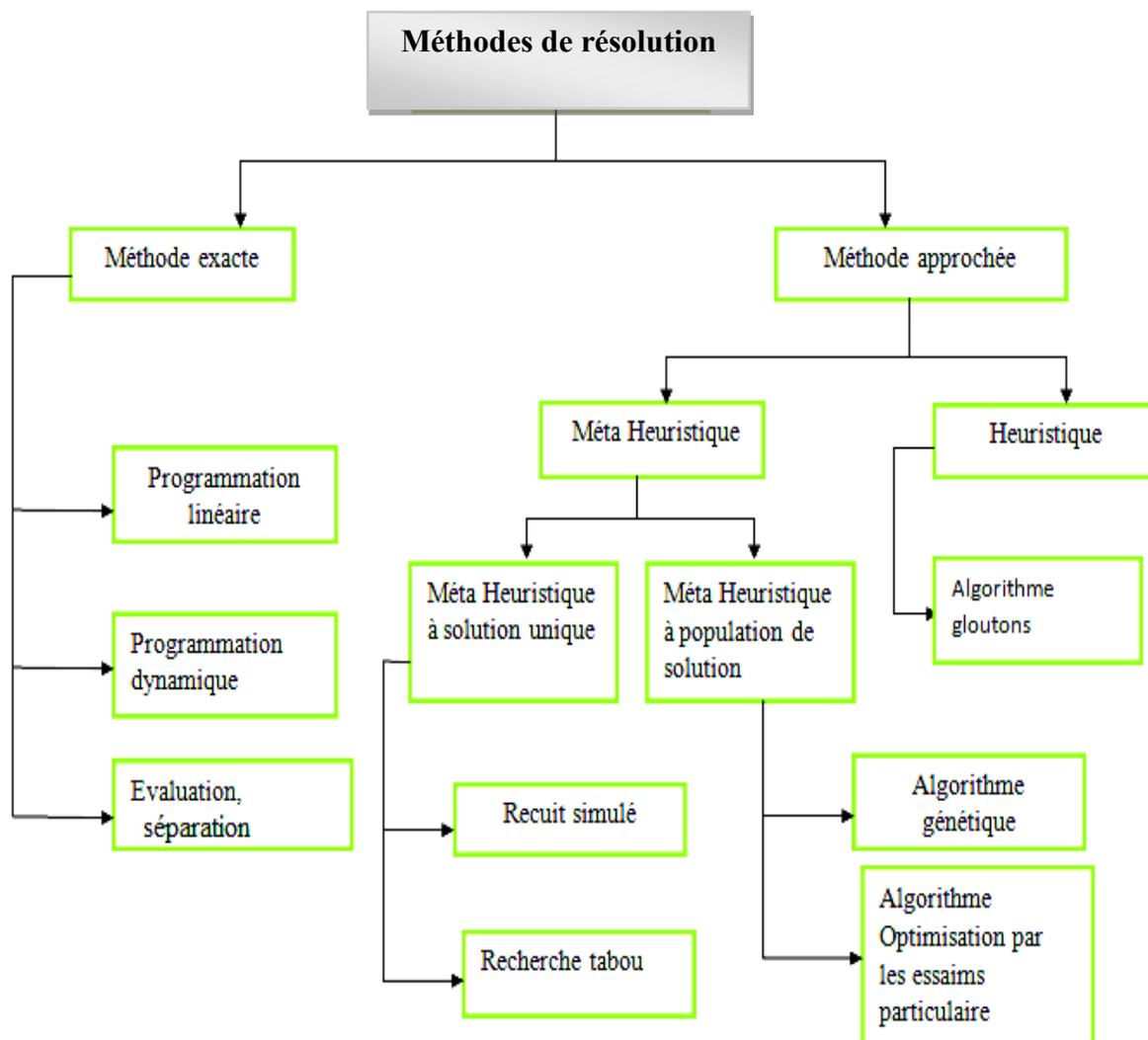


Figure 15: Taxonomie des méthodes de résolution des problèmes d'optimisation.[27]

Dans notre projet on va s'intéresse à l'algorithme d'optimisation par les essais particuliers en raison de sa facilité et de son utilisation fréquente.

5 Fonction Objectif :

La fonction f est connue sous ce nom (on l'appelle aussi fonction de coût, critère d'optimisation ou fitness). La méthode d'optimisation devra optimiser cette fonction (trouver un optimum).[28]

6 Algorithme Optimisation par les essais particulaire (PSO)

6.1 Principe générale :

L'optimisation par essaims de particules (PSO) est un algorithme évolutionnaire qui développe une solution optimale à partir d'une population de solutions candidates. Russel Eberhart (ingénieur électrique) et James Kennedy (psychologue social) ont proposé cet algorithme en 1995. Il a été inspiré par le monde vivant, à savoir le comportement social des animaux qui ont évolué en essaims, comme les bancs de poissons et des nuées d'oiseaux.[24] (**voir Figure 16**) En effet, on peut observer des dynamiques de déplacement assez sophistiquées et complexes chez ces créatures, alors que chaque individu a une "intelligence" limitée et n'a qu'une compréhension locale de la situation dans l'essaim. [24]



(a)



(b)

Figure 16 : (a) Nuage d'oiseaux[29] (b) bancs de poissons[30] .

En raison de sa vitesse de convergence rapide pour l'optimisation à objectif unique, PSO est devenu l'une des méthodes d'optimisation recommandées.[31]

CHAPITRE 3 : Algorithme d'optimisation par les essais particuliers PSO

Un individu de l'essaim ne connaît que la position et la vitesse de ses voisins les plus proches. Pour déterminer son propre déplacement, chaque individu utilise non seulement sa propre mémoire, mais aussi la connaissance locale de ses voisins les plus proches.[28]

Les particules interagissent les unes avec les autres dans l'espace de recherche afin de résoudre le problème et de bénéficier de leur expérience combinée. Au départ, les particules ne savent pas où se trouve l'optimum, elles doivent donc voler à travers l'espace de recherche. Le processus de recherche est basé sur deux principes : [32]

1. **La loi de la communication** : Ils doivent communiquer la mesure développée par chaque particule à tout autre agent présent dans l'essaim. [32]
2. **La loi de l'apprentissage** : Lorsque les particules communiquent leurs valeurs, elles peuvent collaborer et apprendre que le placement de toute particule est supérieur, ce qui leur permet de découvrir la valeur optimale pour l'ensemble de l'essaim (appelée minimum global).[32]

Chaque membre de l'essaim est appelé particule. Une population est formée par la collection de particules. Celle-ci est choisie au hasard dans l'espace de recherche de la fonction objectif à optimiser. Le déplacement d'une particule est déterminé par trois facteurs:[31](voir figure 17)

- a) **Une composante d'inertie** : La particule se déplace dans la même direction et à la même vitesse (suivre sa direction courante de déplacement).[31]
- b) **Une composante cognitive** : La particule a tendance à revenir à un endroit antérieur qui correspond à la meilleure position par laquelle elle est passée précédemment. [31]
- c) **Une composante sociale** : La particule gravite vers la meilleure région traversée par l'ensemble de l'essaim.[31]

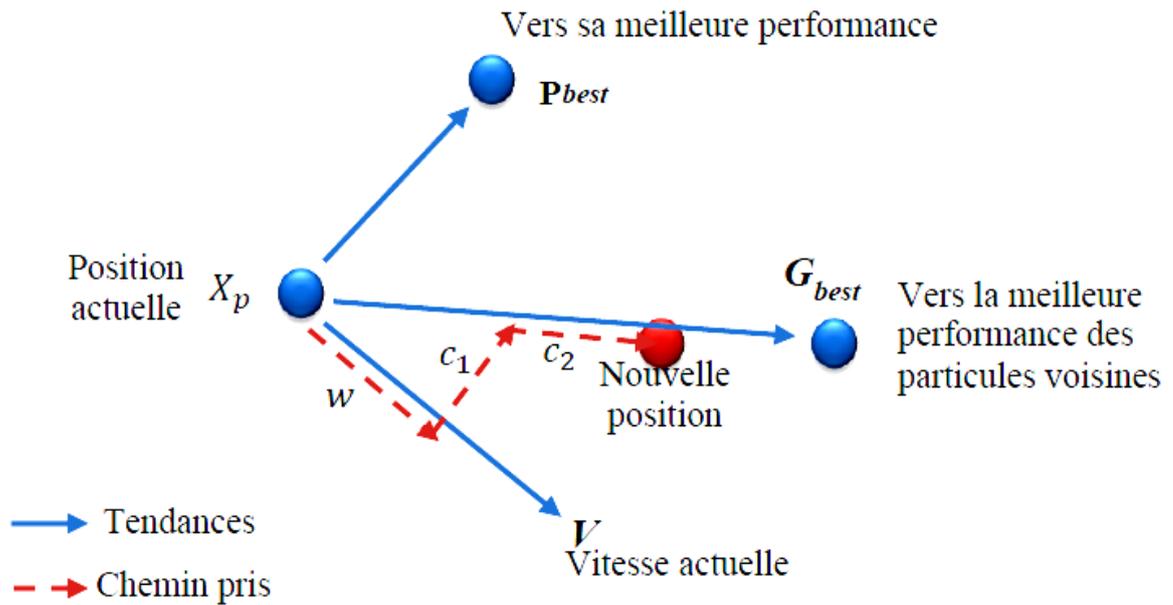


Figure 17 : Déplacement d'une particule.[33]

6.2 Formalisation Mathématique :

Chaque particule est caractérisée par sa position $\vec{X}_i^t = (X_{i1}, X_{i2}, \dots, X_{iN})$ et sa vitesse $\vec{V}_i^t = (V_{i1}, V_{i2}, \dots, V_{iN})$. À chaque itération, ces particules mettent à jour leur vitesse pour se déplacer vers leur meilleur emplacement, qui correspond à la position idéale, en fonction de leurs expériences antérieures et de celles des particules voisines, comme indiqué dans les équations (3), (2) :[31]

$$X_{i,j}^{t+1} = X_{i,j}^t + v_{i,j}^{t+1}, j \in \{1, 2, \dots, N\} \dots\dots\dots (2)$$

$$V_i^{t+1} = \underbrace{\omega V_i^t}_{\text{Vitesse actuelle de la particule}} + \underbrace{C_1 R_1 (P_{i,best}^t - X_i^t)}_{\text{Historique propre à la particule } i} + \underbrace{C_2 R_2 (G_{best}^t - X_i^t)}_{\text{Historique des voisins (essaim)} \dots\dots\dots (3)$$

Tel que :

- ω : appelée coefficient/pourcentage d'inertie. Est une constante, Cependant, dans certains cas, une grande valeur de w correspond à une grande amplitude de mouvement et donc à une exploration globale de l'espace de recherche. Selon [11], w entre 0.8 et 1.2 a une

CHAPITRE 3 : Algorithme d'optimisation par les essais particuliers PSO

meilleure convergence. Son objectif est de trouver un équilibre entre la recherche locale (exploitation) et la recherche globale (exploration). [12]

- **N** : est le nombre de populations, **i** : est le nombre de variables d'optimisation et **t** : est le numéro d'itération.[31]
- **C1** est une constante positive de la pondération dite d'accélération de la composante cognitive et **C2**, c'est une constante d'accélération de la composante sociale. En général, ils prennent des valeurs positives définies dans l'intervalle [0,2].[31]
- **R1** et **R2** sont des valeurs aléatoires entre [0,1] pour apporter un caractère stochastique l'algorithme.[32]
- **P_{best}** : Est la meilleure position de la particule et **G_{best}** : est la meilleure position de tout l'essaim. [31]
- **X** est la position et **V** est la vitesse de particule. [31]

Comme le montre l'équation suivante, la position des particules dans l'espace de solution au temps t est fonction de la direction actuelle de leur mouvement personnel, de leur vitesse actuelle, de leur meilleure performance personnelle et de la meilleure position des particules qui composent le voisinage de cette particule.[34]

$$X_i^t = f(X_i^{t-1}, V_i^{t-1}, P_{i,best}^t, G_{i,best}^t) \dots\dots\dots (4)$$

- **t** : itération (X_i^t La position de la particule $N^\circ i$ à l'itération **t**.)

6.3 Principe de fonctionnement de l'algorithme PSO :

L'algorithme PSO de base est décrit par la figure ci-dessous :[35]

Initialiser aléatoirement N particule : position et vitesse
Evaluer les positions des particules
Tant que le critère d'arrêt n'est pas atteint faire
 Pour **i** allant de 1 à N faire
 Déplacer les particules
 Si f_{xi} Meilleur f_{Pbesti} alors

CHAPITRE 3 : Algorithme d'optimisation par les essais particuliers PSO

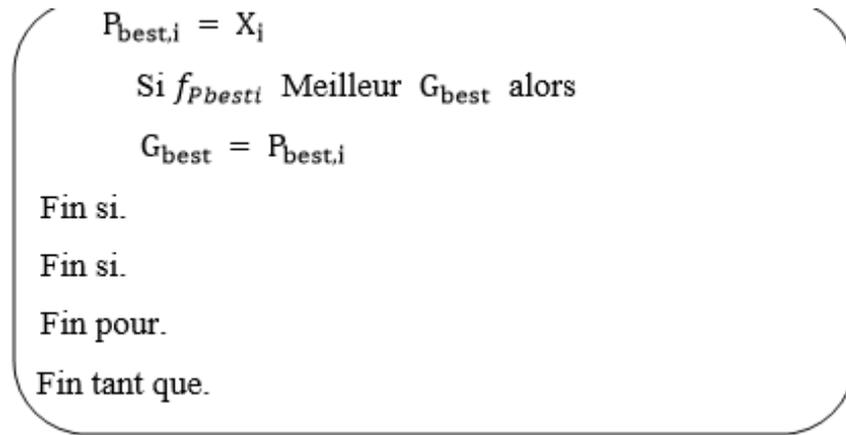


Figure 18 : fonctionnement de l'algorithme PSO.

Le principe de base de l'algorithme consiste à déplacer ces particules dans l'espace de recherche afin de découvrir la meilleure réponse. Au début de la procédure, un essaim est dispersé de manière aléatoire dans l'espace de recherche, chaque particule ayant une vitesse aléatoire.[28]

Cet algorithme concentré sur trois étapes essentielles :

Etape (1) : Initialisation de population (nombre de particule N)

- Initialisé position (x) et vitesse (V)
- Assigné P_{best} et G_{best} (basé sur la fonction objective)

Etape (2) : Mise à jour

- Pour chaque particule :

$$\text{La vitesse : } V_i^{t+1} = \omega V_i^t + C_1 R_1 (P_{i,best}^t - X_i^t) + C_2 R_2 (G_{best}^t - X_i^t)$$

$$\text{La position : } X_{i,j}^{t+1} = X_{i,j}^t + v_{i,j}^{t+1}, j \in \{1, 2, \dots, N\}$$

- Fonction de fitness f_{xi} (fonction objective)
- Mise à jour de $P_{best,i}$ et G_{best}

$$P_{best,i} = X_i \text{ si } f_{xi} \text{ Meilleur } f_{Pbest,i}$$

$$G_{best} = P_{best,i} \text{ si } f_{Pbest,i} \text{ Meilleur } G_{best}$$

Etape (3) : Résiliation

- Les étapes de l'algorithme PSO sont répétées de manière itérative jusqu'à ce que le nombre maximal de générations soit atteint ou qu'un critère de fin soit satisfait.

CHAPITRE 3 : Algorithme d'optimisation par les essais particuliers PSO

- **Convergence** : est le cas où les positions de toutes les particules convergent vers le même ensemble de valeurs, la méthode est supposée avoir convergé.

Finalement, on peut résumer le fonctionnement de l'algorithme par l'organigramme suivant :

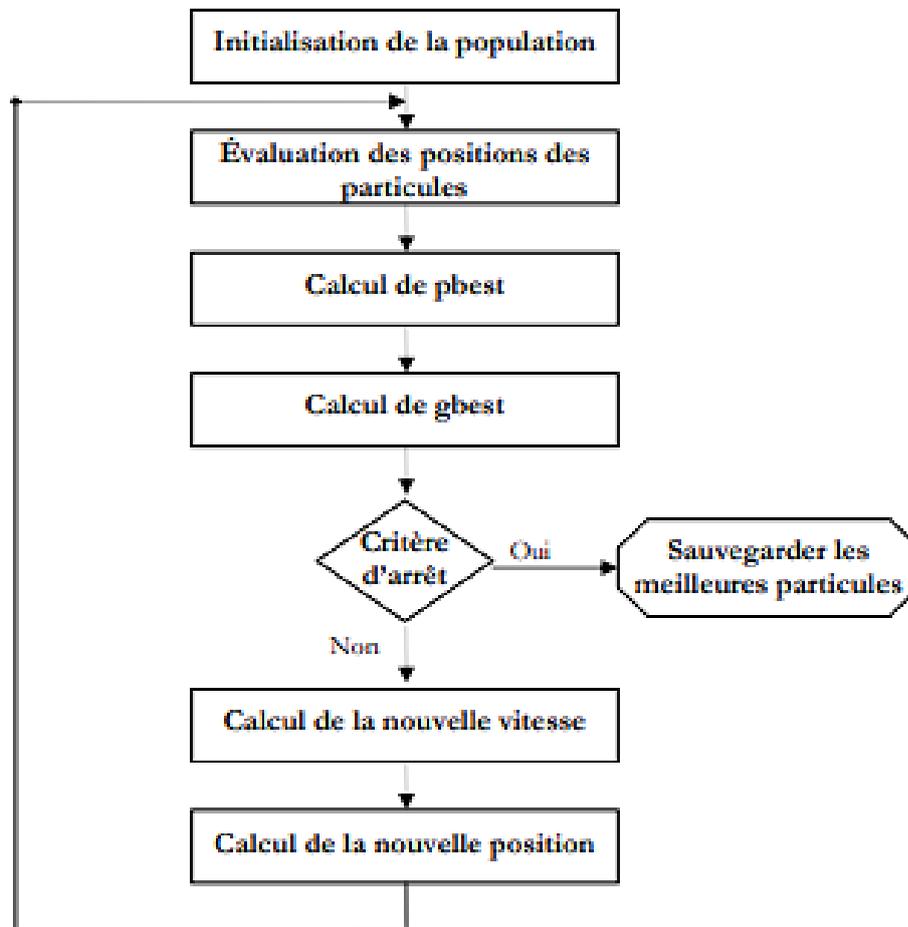


Figure 19 : Organigramme de principe de PSO.[36]

6.4 Avantages de PSO:[28]

- ✓ Des mises en œuvre simples, avec un minimum de paramètres
- ✓ Polyvalence : peut s'attaquer à un large éventail de problèmes
- ✓ Peuvent arriver rapidement à des solutions appropriées

6.5 Inconvénients de PSO: [28]

- ✓ Dans un cadre idéal, il y a une tendance à la convergence rapide et prématurée.
- ✓ Dans la phase de recherche détaillée, la convergence est lente (faible capacité de recherche locale).

- ✓ Sélection des paramètres d'optimisation, qui sont souvent spécifiques au problème.

7 Conclusion

Dans ce chapitre nous avons présenté les méthodes approchées qui ont pour but de trouver une solution admissible en un temps raisonnable, et aussi bien l'algorithme d'optimisation par essaim particulier (PSO) inspiré du monde des animaux (espèces d'oiseaux).

On a compris dans ce chapitre le fonctionnement de l'algorithme PSO qu'on va utiliser dans notre approche et qu'on va implémenter dans une plateforme E-Learning.

CHAPITRE 4 :
**Conception et spécification de
l'approche.**

1 Introduction :

Fournir un apprentissage à distance efficace et d'assurer une certaine performance par rapport au temps de réponse des plateformes d'enseignement à distance en tenant compte des vitesses variantes de connections de utilisateurs, est i considéré comme l'un des grands challenges des plateformes E-Learning.

L'objectif de notre projet est de développer une plateforme d'enseignement à distance auto-adaptative qui permet d'adapter l'utilisation des différentes fonctionnalités proposées par la plateforme à la vitesse de connexion de l'utilisateur à un moment donné.

Nous nous s'intéressons dans ce chapitre à la présentation de notre étude de cas (plateforme d'apprentissage en ligne) et à la conception de notre approche de la planification de l'autoadaptation, En se basant sur la métaheuristique de population (L'algorithme d'Optimisation par les essais particulière (**PSO**) précédemment étudié).

2 Etude de cas :

Comme étude de cas, nous avons choisi une plateforme d'apprentissage en ligne car elle intègre plusieurs services (technologies) pour aider les utilisateurs à mieux apprendre. La plate-forme fournit la liste suivante de services configurables :

- ✓ Accès aux cours sous diverses formes (vidéo, image, texte).
- ✓ Les enseignants et les élèves peuvent échanger des communications de différentes manières (texte, image, vidéo, audio).
- ✓ L'utilisateur a la possibilité de donner son avis sur le cours (commentaire) auquel il s'est inscrit.
- ✓ Donner à l'utilisateur la possibilité de télécharger des fichiers.
- ✓ Les statistiques sur la progression du cours de l'utilisateur.
- ✓ Les enseignants utilisent le streaming vidéo pour dispenser des cours en direct ou pour répondre aux questions des utilisateurs.
- ✓ Personnalisation de la plate-forme en fonction des choix de l'utilisateur, tels que les couleurs de la plate-forme, la taille du texte, la langue utilisée, etc.
- ✓ L'apprenant peut répondre à des quizz pour tester son niveau d'apprentissage.

Cette plateforme peut être adaptée au :

- Vitesse de connexion (par exemple, utiliser cours avec texte par ce que la vitesse de connexion est faible).
- Préférences de l'utilisateur (par exemple, un cours avec vidéo est préférable qu'avec image).
- Consommation des données de connexion.

3 Modélisation des variations dans la plateforme proposée

Nous avons utilisé le modèle de fonctionnalités pour représenter tous les différents types de services disponibles sur notre palteforme(voir **figure 20**). Un modèle de fonctionnalité est une représentation concise de tous les produits d'une ligne de produits logiciels en termes de "fonctionnalité" dans le développement de logiciels. Une fonctionnalité est un attribut du système qui est important pour certaines parties prenantes et qui est utilisé pour capturer des similitudes ou distinguer des systèmes.[37]

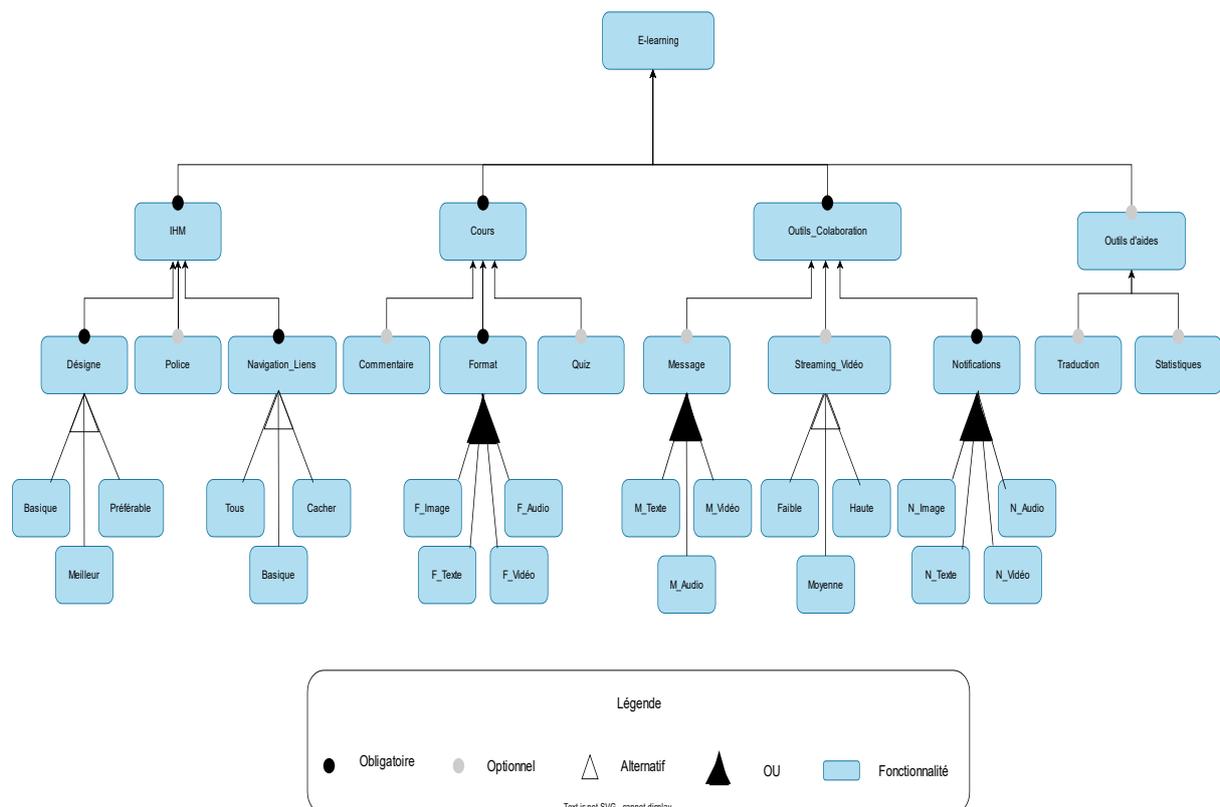


Figure 20: Modélisation des variations

4 Modélisation de notre système

Nous avons choisi le langage de modélisation unifié pour développer notre système (UML). Ce dernier est un langage de modélisation graphique construit sur un méta-modèle qui définit les composants de modélisation (concepts manipulés par le langage) et leurs significations (définitions et sens de leurs utilisations). C'est un langage formel construit autour des diagrammes qui seront produits dans la suite de ce chapitre.

4.1 Diagramme des cas d'utilisation

Le diagramme de cas d'utilisation est un diagramme UML utilisé pour donner une vision globale du comportement fonctionnel d'un système logiciel. Un cas d'utilisation représente une unité discrète d'interaction entre un utilisateur (Human ou Machine) et un système. Il est une entité significative de travail. [38]

Dans un diagramme de cas d'utilisation il existe des acteurs (actors) qui interagissent avec des cas d'utilisation (use case) UC.

Les use case permettent de structurer les besoins des utilisateurs et les objectifs du système. [38]

Dans notre plateforme, un utilisateur peut jouer simultanément les rôles d'apprenant et d'enseignant dans des cours distincts. Par exemple, l'utilisateur X est un élève du cours A qui a également conçu le cours B, il est donc aussi un enseignant du cours B. (**voir figure 21**)

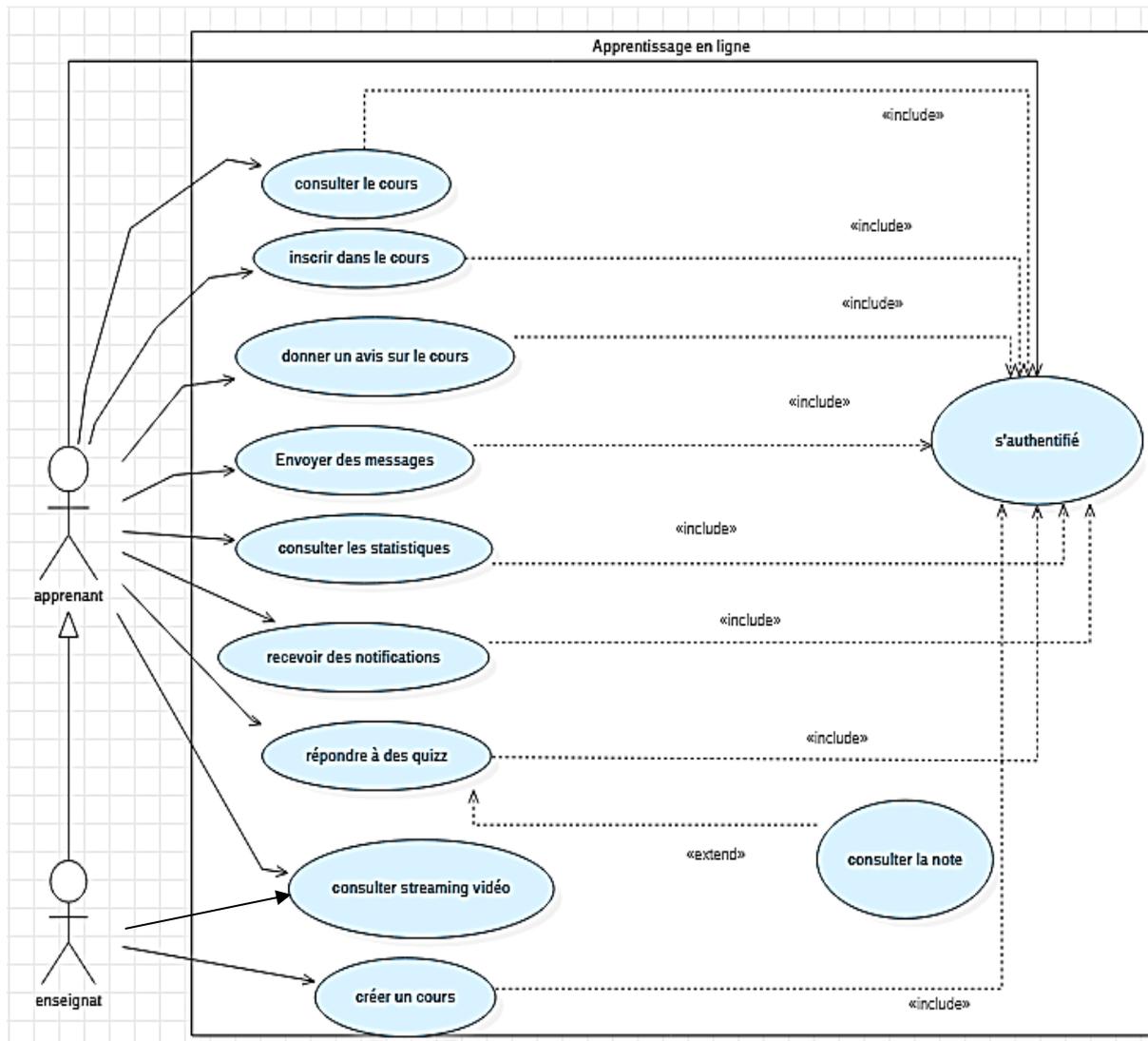


Figure 21: Diagramme de cas d'utilisation.

4.2 Diagramme de classe

Un diagramme de classes représente les nombreux types de connexions statiques qui existent entre les objets ou les données du système. Dans le développement orienté objet, le diagramme de classes est souvent considéré comme le diagramme le plus important. Il définit l'architecture conceptuelle du système : il explique les classes que le système emploie, ainsi que les liens, qui reflètent soit une imbrication mentale (héritage), soit une relation (agrégation).[39] (**voir figure 22**)

CHAPITRE 4 : Conception et spécification de l'approche

Par rapport au **temps de réponse (ms)** car pour nous on veut diminuer le temps de réponse d'utilisateur dans la plateforme et pour avoir ce temps on doit connaitre :

- Premièrement le temps de réponse de serveur de notre plateforme pour l'obtention du service (l'achèvement de la requête).
- Deuxièmement le temps de réponse de connexion d'utilisateur qui est connu sous le nom **Latence** « en anglais **Latency** » de l'utilisateur qu'on peut connaître avec un test de connexion⁸.

Donc comme la montre la figure ci-dessous, le temps de réponse d'un utilisateur dans notre plateforme (500ms) **égale** leur temps de réponse de connexion ou **latence** (400 ms) **plus** le temps de réponse de serveur de notre plateforme (100ms).

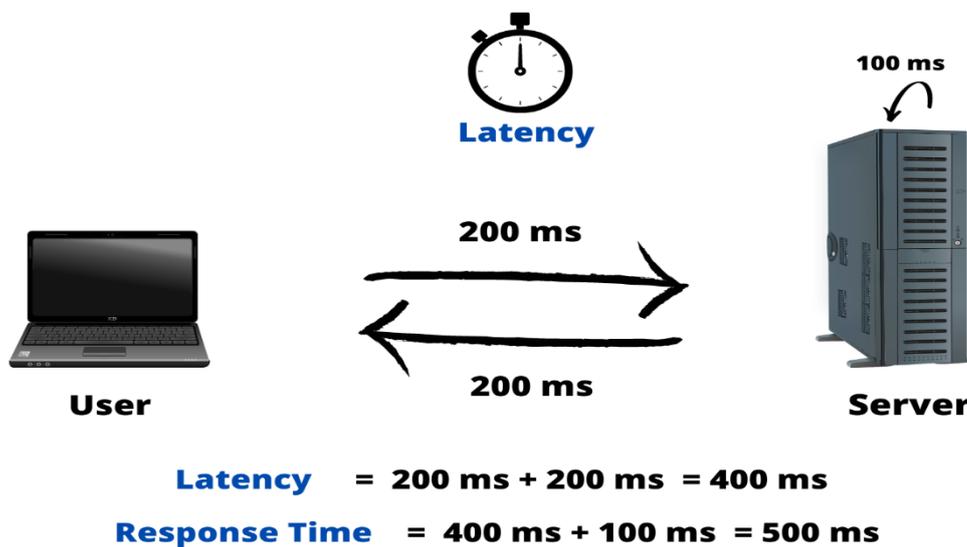


Figure 23: Représentation de latence.

6 Déclenchement de notre approche

Le critère principal qui détermine le début de notre approche est la vitesse de connexion de l'utilisateur, plus précisément son temps de réponse dans la plateforme avec cette vitesse, donc si l'utilisateur a un bon temps de réponse (ce qui implique une bonne vitesse de connexion), il

⁸ Accessible sur : <https://testmy.net/latency>

CHAPITRE 4 : Conception et spécification de l'approche

peut utiliser tous les services avec les meilleures variations, sinon dans le cas contraire, notre approche va déclencher (faible temps de réponse qui implique une faible connexion).

La vitesse de connexion est déterminée par trois facteurs principaux :

- **Le débit descendant (ou téléchargement, réception)⁹**: cette mesure fait référence à la vitesse de téléchargement de l'Internet, c'est-à-dire la vitesse à laquelle les données sont transmises du réseau à votre ordinateur. L'affichage des pages en ligne et le téléchargement des données (images, audio, dossiers, etc.) dépendent tous deux de cette vitesse "descendant".
- **Le débit montant (ou upload, envoi)¹⁰**: Identique au téléchargement, mais dans le sens inverse, c'est-à-dire la vitesse de l'internet depuis votre machine vers le réseau.
- **Latence (ping)¹¹**: Il mesure le temps nécessaire à deux ordinateurs reliés à un réseau pour réagir. Plus le temps est court, plus la connexion est rapide.

Nous sommes intéressés par le temps de réponse pour le déclenchement de notre approche, et le facteur qui représente cela dans la vitesse de connexion est la latence (délai) de la connexion de l'utilisateur, et plus la latence est petite, plus le temps de réponse est petit, donc si l'utilisateur a une grande latence, c'est-à-dire son temps de réponse est grand, alors notre approche doit commencer (la limite du temps de réponse de l'utilisateur dans la plate-forme qui ne doit pas dépasser est de 4000ms dans notre cas).

Pour ce faire, nous avons mis au point une technique qui compare la latence de connexion de l'utilisateur au temps de réponse des services activés dans notre plateforme et détermine si notre approche doit être utilisée ou non.

Nous avons donc un observateur qui exécute une fonction lorsque la valeur de latence de l'utilisateur change ; cette fonction calcule le total du temps de réponse (TR) de tous les services activés dans la plate-forme, plus la latence de l'utilisateur pour chaque service activé. Si le résultat dépasse un certain seuil, notre technique commencera à fournir une stratégie idéale qui réduit le temps de réaction ; sinon, rien ne sera fait. La stratégie peut être représentée comme suit :

⁹Accessible sur :<https://www.echosdunet.net/adsl/test>

¹⁰Accessible sur :<https://www.echosdunet.net/adsl/test>

¹¹Accessible sur :<https://www.echosdunet.net/adsl/test>

Si $(\sum E(i) (TR(i) + latence) > \text{seuil})$

Alors Approche Activé

Sinon rien à faire

- **TR(i)** : Temps de réponse de serveur de plateforme pour le service i.
- **E(i)** : L'état de service i, prendre les valeurs 0 si désactivé et 1 si activé.
- **Latence** : Temps de réponse de connexion d'utilisateur.
- **Seuil** : Dans notre cas, le niveau du temps de réaction d'un utilisateur sur notre plateforme ne doit pas dépasser 4000ms. Lorsqu'un utilisateur se connecte à notre plateforme pour la première fois et lorsque la vitesse de connexion de l'utilisateur change, notre stratégie exécute et réalise ce test.

7 Fonctionnement de l'approche

En cas de connexion faible, nous utiliserons des techniques d'optimisation par essaims de particules (PSO) pour élaborer un plan d'adaptation optimal (la meilleure stratégie d'adaptation). Notre défi peut être décrit comme un problème d'optimisation (minimisation du temps de réponse et de la consommation de données avec maximisation de l'utilité des fonctionnalités). Ainsi, si la connexion est mauvaise, notre méthode génère un plan d'adaptation qui inclut une variation de chaque service. A titre d'exemple, un plan peut être :

- Cours : Texte.
- Design : Basique.
- Lien de navigation : Tous.
- Message : Audio.
- Téléchargement : Activé.
- Police : Grande.
- Commentaire : Activé.
- Statistique : Désactivé.
- Streaming vidéo : qualité moyenne.
- Quiz : Activé.
- Traduction : Activé.

7.1 But de notre approche

Notre objectif est de réduire le temps de réaction de la plate-forme et la consommation de données de connexion tout en augmentant ou en réduisant la qualité du service afin de maximiser l'utilité du service en fonction de la vitesse de connexion des utilisateurs.

7.2 Tableau de variations

Ce tableau répertorie tous les services et variantes disponibles sur notre plateforme, ainsi que leur temps de réponse (ms), leur utilité et leur consommation de données de connexion (Kbps).

(Voir tableau 2 et 3)

Tableau 2:Tableau de variations

Table de variations 

	Variation	Temps de Réponse (Ms)	Utilité	Consommation (Kbts)
Cours	Vidéo	400	700	500
	Photo	200	150	200
	Texte	100	30	100
Design	Meilleur	500	80	300
	Préférable	300	60	150
	Basique	100	30	100
Navigation des liens	Tous	200	80	200
	Méta	100	50	130
	Caché	0	20	10
Quiz	Activé	200	150	120
	Désactivé	0	0	0
Message	Désactivé	0	0	0
	Vidéo	400	70	500
	Texte	100	20	150
	Audio	200	50	200

Tableau 3:Tableau de variations (suite)

Streaming	Traduction	Activé	250	80	200
		Désactivé	0	0	0
Commentaire		Activé	200	80	200
		Désactivé	0	0	0
vidéo		Désactivé	0	0	0
		Haute	600	70	600
		Moyenne	400	60	400
		Faible	200	30	200
Téléchargement		Activé	150	90	50
		Désactivé	0	0	0
Statistique		Activé	200	80	200
		Désactivé	0	0	0
Police		Désactivé	0	0	0
		grande	150	40	40
		petite	50	10	20

7.3 Spécification de l'algorithme PSO pour notre approche

L'algorithme PSO est un algorithme d'optimisation basé sur une population initial appelée un essaim, qui représente un ensemble de particules, dont chaque particule a une position X, Une vitesse V et Pbest, comme nous allons expliquer dans le chapitre précédent. L'application de PSO dans notre approche, consiste à choisir une particule qui représente une solution parmi beaucoup, c'est-à-dire un état de plateforme à un instant donnée. Dans la première itération, la

CHAPITRE 4 : Conception et spécification de l'approche

position X représente la fitness de particule P_i et V la valeur de la vitesse de cette particule, qui a été choisi aléatoirement, P_{best} c'est la fitness de la particule et G_{best} représente la meilleure position dans l'ensemble initial des particules, qui a été choisi aussi de façon arbitraire.

7.4 Codage de l'essaim particulaire

La particule (individu) est un plan de solution possible pour notre approche, L'APSO utilise plusieurs particules dans leur fonctionnement, c'est pour ça le codage est l'une des étapes importantes pour l'implémentation de cet algorithme. Une particule constituée de services (dans notre cas) de plateforme (cours avec image, quiz activé, ...).

Le service se caractérise par :

1. **Un état** : c'est si le service est activé ou désactivé.

Tableau 4: Codage des états.

Etat	Activé	Désactivé
Codage	1	0

2. **Une utilité** : qui le spécialise, l'utilité c'est-à-dire l'efficacité de l'utilisation de ce service. Par exemple l'utilité de cours avec image est meilleure que l'utilité de cours avec texte à cause de son efficacité.
3. **Une consommation des données** : c'est à dire combien le service consomme les données de connexion.
4. **Un temps de réponse** : Temps de réponse de service.

La particule est formée en collectant simplement tous les services.

7.5 Descriptions des services :

Dans notre approche un service peut avoir plusieurs variations, par exemple le cours peut être : vidéo, image et texte, Pour garder la structure correcte d'un individu et éviter de tomber dans une configuration qui n'a aucun sens (ex : l'approche donne une solution de cours avec vidéo et avec image), nous avons effectué un classement des services et un codage d'état des services de la manière suivante : Les variations des services sont choisies d'une manière aléatoire.

Service 1 : Cours

CHAPITRE 4 : Conception et spécification de l'approche

Tableau 5: Codage de service de cours

Etat	Vidéo	Image	Texte
Codage	0	1	2

Service 2 : Design

Tableau 6: Codage de service de Design

Etat	Meilleur	Préférable	Basique
Codage	0	1	2

Service 3 : Navigation_Liens

Tableau 7: Codage de service de Navigation_Liens

Etat	Tous	Méta	Caché
Codage	0	1	2

Service 4 : Message

Tableau 8: Codage de service de Message

Etat	Désactivé	Vidéo	Audio	Texte
Codage	0	1	2	3

Service 5 : Téléchargement

Tableau 9: Codage de service de Téléchargement

Etat	Activé	Désactivé
Codage	1	0

Service 6 : Police

CHAPITRE 4 : Conception et spécification de l'approche

Tableau 10: Codage de service de Police

Etat	Désactivé	Grande	Petite
Codage	0	1	2

Service 7 : Statistique

Tableau 11: Codage de service de Statistique

Etat	Activé	Désactivé
Codage	1	0

Service 8 : Commentaries

Tableau 12: Codage de service de Commentaire

Etat	Activé	Désactivé
Codage	1	0

Service 9 : Streaming video (Qualité)

Tableau 13: Codage de service de Streaming vidéo (Qualité)

Etat	Désactivé	Haute	Moyenne	Faible
Codage	0	1	2	3

Service 10 : Traduction.

Tableau 14: Codage de service de Traduction.

Etat	Activé	Désactivé
-------------	---------------	------------------

CHAPITRE 4 : Conception et spécification de l'approche

Codage	1	0
--------	---	---

Service 11 : Quiz

Tableau 15: Codage de service de Quiz.

Etat	Activé	Désactivé
Codage	1	0

7.6 La population initiale :

L'essai initial est considéré comme un point de départ pour l'APSO et on l'appelle la première population. Les essaims futurs se développent en fonction d'elle pour atteindre la meilleure solution avec l'utilisation d'opérateurs (Mouvement de l'ensemble des particules : la position (X_i) et vitesse (V_i)).

Un bon départ facilite considérablement le développement du processus. Il existe différentes techniques pour créer une population de départ. Les techniques les plus connues sont le tirage aléatoire, l'heuristique, ou un mélange de solutions heuristiques et d'aléatoire. Dans notre technique, la population de départ est produite de manière aléatoire, et l'état de service est géré par la fonction "Random", qui produit des fluctuations aléatoires.

Voici un exemple de population de 4 particules sur 11 services.

Tableau 16: Exemple d'une population

Services	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11
Particule 01	1	2	1	3	0	2	0	1	1	1	0
Particule 02	0	1	0	2	0	1	0	1	0	1	1
Particule 03	2	0	2	3	1	2	1	0	2	0	0
Particule 04	1	2	1	1	1	1	0	1	3	1	1

Donc par exemple la description de la particule 02 est :

- **Service 1** (cours) : 0 => vidéo.
- **Service 2** (design) : 1 => Préférable.
- **Service 3** (Liens de navigation) : 0 => Tous.

- **Service 4** (Message) : 2 =>Audio.
- **Service 5** (Téléchargement) : 0 => Désactivé.
- **Service 6** (Police) : 1 =>Grande.
- **Service 7** (Statistique) : 0 => Désactivé.
- **Service 8** (Commentaire) : 1 => Activé.
- **Service 9** (Streaming vidéo) : 0 => Désactivé.
- **Service 10** (Traduction) : 1 => Activé.
- **Service 11** (Quiz) : 1 =>Activé.

7.7 La fonction de fitness

Nous avons besoin d'une fonction appelée fonction de fitness pour découvrir la meilleure solution dans l'espace des solutions (trouver l'individu le plus proche de l'optimal). Pour déterminer l'optimalité de chaque individu, cette dernière emploie des critères qui spécialisent notre problème d'optimisation. Dans notre système nous sommes concentrés sur les trois critères suivants :

- Critère 1 : la minimisation de temps de réponse.
- Critère 2 : la maximisation d'utilité.
- Critère 3 : la minimisation de la consommation de données.

La fonction de fitness $f(X)$ est définie comme la somme pondérée des trois objectifs (critères) suivants :

- **Temps de réponse TR (i)** : temps de réponse d'un service.
- **L'utilité U (i)** : l'utilité de service ou l'efficacité de service.
- **Consommation C(i)** : consommation de données d'un service.
- **Les poids W_i** : sont affichés sous forme de nombres de types réels liés à chaque critère, qui reflètent leur pertinence par rapport à la demande de l'utilisateur. Le total de leurs poids est égal à 1.

Nous avons choisi uniquement le critère 1 avec une pondération (W_1) de 60% et les critères 2 et 3 avec une pondération de 20%. Parce que nous privilégions la réduction du temps de réaction (réponse) à la maximisation de l'utilité et à la minimisation de la consommation de données.

CHAPITRE 4 : Conception et spécification de l'approche

Tableau 17: les poids des critères

Critères	Poids (W)
Temps de réponse moyen (TR)	60%
Consommation moyen (C)	20%
Utilité moyen (U)	20%

La fonction de fitness $f(X)$ pour la particule X de la variante N est la suivante :

Si l'on veut minimiser un critère, on multiplie par (-1) ; sinon, on multiplie par (+1).

$$f(X) = \sum_{i=1}^N E(i) (-W1 * Tri - w2 * Uti + W3 * Ci) \dots\dots\dots 1$$

Nous pouvons découvrir des valeurs d'aptitude négatives dans cette fonction, ce qui pourrait poser des problèmes avec l'implémentation de l'algorithme. Ainsi, pour éviter d'avoir une valeur de fitness négative, nous allons normaliser en utilisant une formule, La formule est expliquée en détail dans la section suivante.

7.8 Matrice de limites :

Nous allons créer une matrice des limitations du ML en utilisant des critères de qualité. Cela fournit une marge de valeurs acceptables pour chaque critère et son poids. Chaque colonne représente un critère, avec la valeur minimale admissible, la valeur maximale acceptable et, enfin, son poids.

$$ML = \begin{pmatrix} Q_{Tr}^{min} & Q_{Ut}^{min} & Q_C^{min} \\ Q_{Tr}^{max} & Q_{Ut}^{max} & Q_C^{max} \\ W_{Tr} & W_{Ut} & W_C \end{pmatrix}$$

6.9 La fonction de fitness normalisée :

Nous avons trois critères de normalisation : temps de réponse Q_{Tr} , utilité Q_{Ut} , consommation Q_C . Nous devons normaliser les trois critères de qualité à la même échelle dans [0,1] parce que leurs normes diffèrent.

Supposons que $Q(S) = X(S) \cup Y(S) \cup Z(S)$ où $X(S) = \{Q_{Ut}(S)\}$ un ensemble de critères de qualité positifs et $Y(S) = \{Q_{Tr}(S)\}$, $Z(S) = \{Q_C(S)\}$ 2 ensembles de critères de qualité

CHAPITRE 4 : Conception et spécification de l'approche

négatifs. On note une meilleure qualité de X par des valeurs plus élevées calculées par l'équation (2) et une meilleure qualité de Z et Y par des valeurs plus faibles. Les valeurs des critères de qualité de X (respectivement Y, Z) sont normalisées à X' (respectivement Y', Z') comme :

$$X'_i(S) = \left\{ \frac{X_i(S) - Q_i^{\min}}{Q_i^{\max} - Q_i^{\min}} \right\} i = \text{Ut} \dots 2$$

$$Y'_j(S) = \left\{ \frac{Q_j^{\max} - Y_j(S)}{Q_j^{\max} - Q_j^{\min}} \right\} j = \text{Tr} \dots 3$$

$$Z'_k(S) = \left\{ \frac{Q_k^{\max} - Z_k(S)}{Q_k^{\max} - Q_k^{\min}} \right\} k = \text{C} \dots 4$$

Une simple somme pondérée des valeurs des critères normalisés est utilisée pour déterminer la fonction de fitness f :

$$f(S) = \sum_{\forall i | Q_i \in X} W_i * X'_i(S) + \sum_{\forall j | Q_j \in Y} W_j * Y'_j(S) + \sum_{\forall k | Q_k \in Z} W_k * Z'_k(S) \dots 5$$

6.10 Exemple applicatif :

Pour faciliter les calculs, on a décidé de choisir 2 particules (solutions) avec 3 services (cours, design, streaming vidéo)

Particule 1 : {cours (1), design (2), streaming vidéo (1)}.

Particule 2 : {cours (0), design (1), streaming vidéo (0)}.

6.11 Les étapes de l'algorithme PSO :

- Initialisation paramétrée :

Tableau 18 : Les paramètres initiaux

Paramètre	Valeur
Population	2
Paramètre cognitive C1	0.5
Paramètre cognitive C2	0.5
Paramètre R1	0.1

CHAPITRE 4 : Conception et spécification de l'approche

Paramètre R2	0.1
Poids d'utilité	0.2
Poids de consommation	0.2
Poids de temps de réponse	0.6
Nombre d'itérations	2

Dans notre exemple on a 2 particules avec leurs fitness qui présenté dans le tableau suivant :

Tableau 19: Fitness des particules

Particule	Fitness
P 1	1.27
P 2	1.7

Le fitness de P1 est calculé comme suit :

- **Matrice Limite :**

$$ML = \begin{pmatrix} 0 & 0 & 0 \\ 600 & 700 & 600 \\ 0.6 & 0.2 & 0.2 \end{pmatrix}$$

$$X'(photo) = \left\{ \frac{X_{Ut}(photo) - Q_{Ut}^{min}}{Q_{Ut}^{max} - Q_{Ut}^{min}} \right\} = \frac{150-0}{700-0} = 0.21$$

$$Y'(photo) = \left\{ \frac{Q_{Tr}^{max} - Y_{Tr}(photo)}{Q_{Tr}^{max} - Q_{Tr}^{min}} \right\} = \frac{600-200}{600-0} = 0.67$$

$$Z'(photo) = \left\{ \frac{Q_C^{max} - Z_C(photo)}{Q_C^{max} - Q_C^{min}} \right\} = \frac{600-200}{600-0} = 0.67$$

$$X'(Basique) = 0.04 ; Y'(Basique) = 0.83 ; Z'(Basique) = 0.83$$

$$X'(Haute) = 0.1 ; Y'(Haute) = 0 ; Z'(Haute) = 0$$

$$f(P1) = 0.07 + 0.9 + 0.3 = 1.27$$

CHAPITRE 4 : Conception et spécification de l'approche

$$f(P2) = 0.22 + 1.1 + 0.83 = 1.7$$

- Initialisé P_{best} , G_{best} et $V1$, $V2$ (aléatoirement) :

$P_{best} = X$ (fitness de P1 $f(P1)$) au début.

$P_{best} = X$ (fitness de P2 $f(P2)$) au début

$G_{best} =$ fitness de P2 $f(P2)$

$V1 = 0.4$.

$V2 = 0.2$.

1. Mise à jour :

Itération 1 :

La fonction de vélocité calculée comme suit :

$$V_i^{t+1} = \omega V_i^t + C_1 R_1 (P_{i,best}^t - X_i^t) + C_2 R_2 (G_{best}^t - X_i^t)$$

$$V1' = 0.4 + 0.05 * 0 + 0.05 * 0.43 = 0.42$$

$$V2' = 0.2 + 0.05 * 0 + 0.05 * 0 = 0.2$$

La fonction de position ou nouvelle fitness calculée comme suit :

$$X_{i,j}^{t+1} = X_{i,j}^t + v_{i,j}^{t+1}$$

$$X1' = 1.27 + 0.42 = 1.69$$

$$X2' = 1.7 + 0.2 = 1.9$$

Mise à jour de P_{besti} et G_{best} :

$$P_{best1'} = X1' = 1.69$$

$$P_{best2'} = X2' = 1.9$$

$$G_{best}' = P_{best2'} = 1.9 \text{ (car } P_{best2'} \text{ meilleur que } G_{best} \text{)}$$

Itération 2 :

$$V1'' = 0.43$$

CHAPITRE 4 : Conception et spécification de l'approche

$$V2'' = 1.9$$

$$X1'' = 2.12$$

$$X2'' = 3.8$$

$$P_{best1''} = 2.12 \text{ (car } 2.12 > 1.69)$$

$$P_{best2''} = 3.8 \text{ (car } 3.8 > 1.9)$$

$$G_{best}'' = P_{best2''} \text{ (car } P_{best2''} \text{ meilleur que } G_{best}')$$

Fin de l'algorithme après 2 itérations → Le particule 2 est la meilleure solution.

7 Conclusion :

Dans ce chapitre, nous avons présenté l'étude de cas de notre plate-forme, ainsi que les spécificités de notre approche (algorithme PSO), le codage des particules et la fonction de fitness, et les techniques d'évaluation de notre algorithme. Le chapitre suivant sera consacré à l'exposition des tests et des résultats de nos études.

CHAPITRE 5 :
**Implémentation de la plateforme et
test.**

1 Introduction

Précédemment nous avons présenté notre solution pour l'optimisation de la planification de l'auto adaptation ainsi que les techniques et les mécanismes utilisés pour avoir un bon résultat.

Dans ce chapitre nous allons présenter l'environnement et les outils de travail utilisés pour l'implémentation de notre approche, par la suite décrire les tests et les résultats obtenus.

2 Les outils utilisés

Pour faciliter la réalisation des projets et des approches de fin d'étude. Aujourd'hui, La technologie offre plusieurs outils de développement, on a utilisé certains parmi eux selon nos objectifs et nos besoins.

2.1 Les Systèmes de Gestion de Contenu (CMS)

Le système de gestion de contenu (CMS) est un outil gratuit pour la création et la gestion de sites web. Il dispose de plusieurs fonctionnalités pour la mise en place d'un site web dynamique, notamment une interface d'administration (back-office) qui permet à une personne ne connaissant pas le PHP, le HTML ou d'autres langages de créer, modifier et configurer facilement un site afin d'organiser les différents contenus informatifs.[40]

Il existe aujourd'hui différentes technologies (CMS) qui vous permettent de développer simplement des portails web sans avoir besoin de compétences techniques considérables.

WordPress¹, Drupal², Joomla³, et d'autres en sont des exemples. Ces outils présentent divers avantages :[40]

- Gestion et organisation efficaces du contenu ;
- Développement automatique des menus et des liens de navigation ;
- Administration des utilisateurs et séparation des rôles des utilisateurs ;
- Séparation du contenu et de la conception ;

¹ Accessible sur: [WordPress: Https: //fr.wordpress.org](https://fr.wordpress.org)

² Accessible sur : [Drupal : Http : //drupalfr.org](http://drupalfr.org)

³ Accessible sur : [Joomla : https://www.joomla .fr](https://www.joomla.fr)

• Fonctionnement du CMS

Les principes de base du fonctionnement des systèmes de gestion de contenu sont toujours les mêmes ; tous les progiciels CMS sont conçus en PHP et doivent interagir avec la base de données MYSQL. Tout le matériel modifié est sauvegardé dans une base de données, et les données de la base de données sont accessibles par un interprète PHP[40].

• WordPress

Est un système de gestion de contenu (CMS) gratuit créé en PHP (Mettre un point à un site de vente de vêtements), le plus connu dans le monde du Web, et il est extrêmement simple à installer, à maintenir et à configurer. Il fonctionne bien pour les sites de petite et moyenne taille, quelle que soit la complexité de leur conception. Les utilisateurs de WordPress sont soutenus par un grand groupe de bénévoles[40].

Dans notre travail, on a utilisé **Local by Fly Wheel**¹²(figure ci-dessous), un outil qui s'adresse aux concepteurs WordPress, aux développeurs et à tous ceux qui s'intéressent au développement local. Il s'agit d'une application de développement WordPress locale qui a vraiment tout pour plaire.

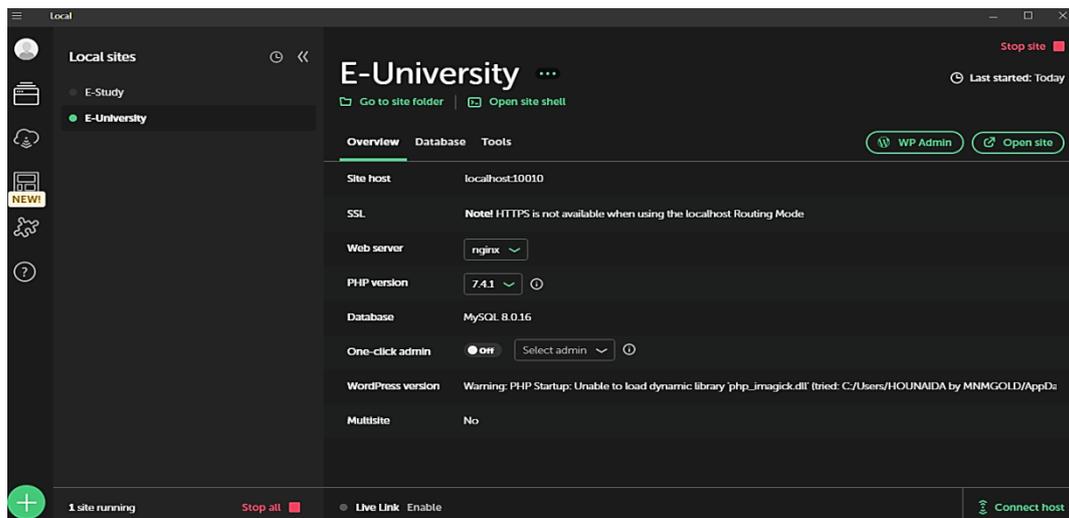


Figure 24: Interface de Local by Flywheel

• Définition d'un thème WordPress

Un thème WordPress est une collection de fichiers qui vont ajuster et personnaliser l'apparence générale du site ainsi que la mise en page du contenu. Par conséquent, un thème influence l'ensemble de l'aspect esthétique du site web. Un thème WordPress, en revanche, ne modifiera

¹²Accessible sur : <https://localwp.com/>

CHAPITRE 5 : Implémentation de la plateforme et test

pas le matériel généré, comme les articles, les titres et les descriptions du site, ...Par conséquent, il est essentiel de séparer la conception graphique d'un site de son contenu.[41]

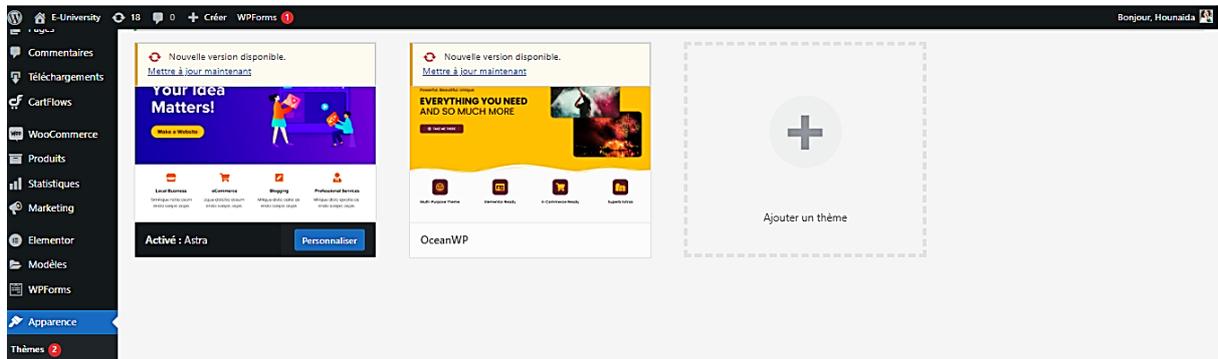


Figure 25: Interface des thèmes WordPress

Le thème utilisé :Astra.

- **Définition d'une extension WordPress**

Une extension (ou plugin en anglais) est **utilisée pour ajouter des fonctionnalités à WordPress**. Il en existe près de **60 000 à ce-jour (majoritairement gratuites)**, dans tous les domaines : référencement, sécurité, e-commerce, sauvegardes, etc.(voir figure ci-dessous)[42]

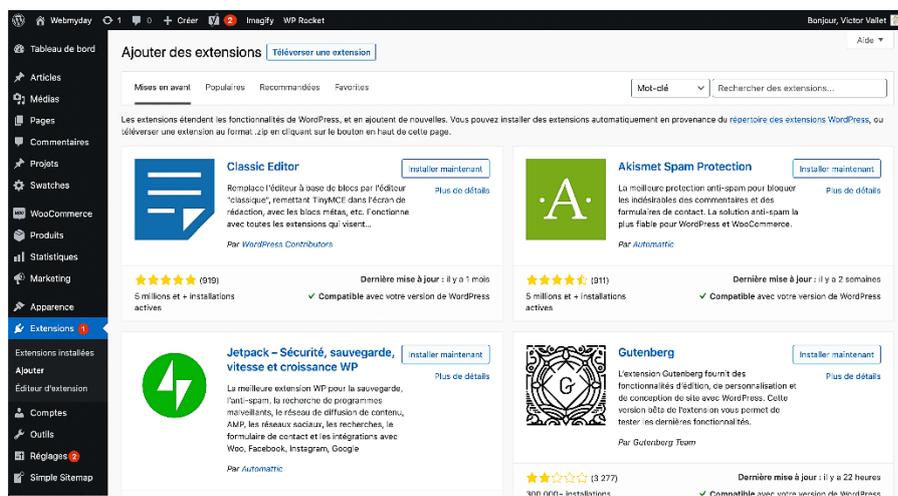


Figure 26 : Interface des extensions WordPress.

Les extensions utilisées :

- ✓ **LearnPress** : C'est l'un des meilleurs plugins WordPress LMS qui peut être utilisé pour créer et vendre facilement des cours en ligne.
- ✓ **Elementor** : C'est un constructeur de pages, accessible aux utilisateurs sous forme de plug-in. **Cet outil est indispensable pour créer des pages complexes sans coder et en s'inspirant de modèles prêts à l'emploi.**

CHAPITRE 5 : Implémentation de la plateforme et test

- ✓ **Google langage translator** : est la solution tout-en-un pour créer un site WordPress multilingue. C'est l'un des moyens les plus rapides et les plus simples de rendre votre site accessible aux personnes qui parlent différentes langues.
- ✓ **Login designer** : comme son nom l'indique, nous permettra de personnaliser notre page de connexion WordPress.
- ✓ **Réseaux sociaux et icônes de partage (Ultimate Social Media)** : nous permet d'ajouter des icônes de partage pour Email, Facebook, Twitter, LinkedIn, Pinterest, Instagram, YouTube, et de télécharger des icônes de partage personnalisées de notre choix.

2.2 L'environnement de développement Apache Netbeans IDE 14

Netbeans IDE est un environnement de développement intégré gratuit et open source pour Windows, Mac, Linux et Solaris. L'EDI facilite la création d'applications en ligne, d'entreprise, de bureau et mobiles pour les plateformes Java et HTML5. Il prend également en charge la création d'applications PHP et C/C++.[43]

2.3 Highcharts

Highcharts¹³ est une bibliothèque graphique basée sur JavaScript. Elle vous permet de dessiner tout type d'image de manière simple. Elle est compatible avec tous les navigateurs et appareils contemporains. Elle est open-source et gratuite. Elle permet d'afficher des infobulles sur les axes et les courbes, ce qui est assez pratique. Il est possible d'exporter les données.

En utilisant le plugin correspondant, les données peuvent être immédiatement saisies. Il est également possible d'importer des données d'une autre source. Le site officiel propose également une multitude de contenus, dont des exemples et de nombreuses formes d'assistance.[44]

3 Présentation de notre plateforme

Dans notre projet de fin d'étude on a une plateforme E-Learning qui implémente une approche d'adaptation, on va présenter premièrement les interfaces et les services (fonctionnalités) de notre plateforme, après on va présenter les interfaces de la simulation qu'on a fait pour tester notre approche.

¹³Accessible sur : <https://www.highcharts.com/>.

3.1 Présentation les services (fonctionnalités) de notre plateforme :

Dans notre plateforme on a plusieurs interfaces qui contiennent plusieurs fonctionnalités offertes à utilisateurs de notre plateforme, donc les interfaces présentées suivantes sont les interfaces essentielles et les fonctionnalités qu'on a implémentées.

- **Interface d'authentification**

L'Interface d'authentification c'est la première interface de notre plateforme, qui va être en face de l'utilisateur, dont il doit s'authentifier, pour accéder aux différents services de plateforme, comme le montre cette figure :

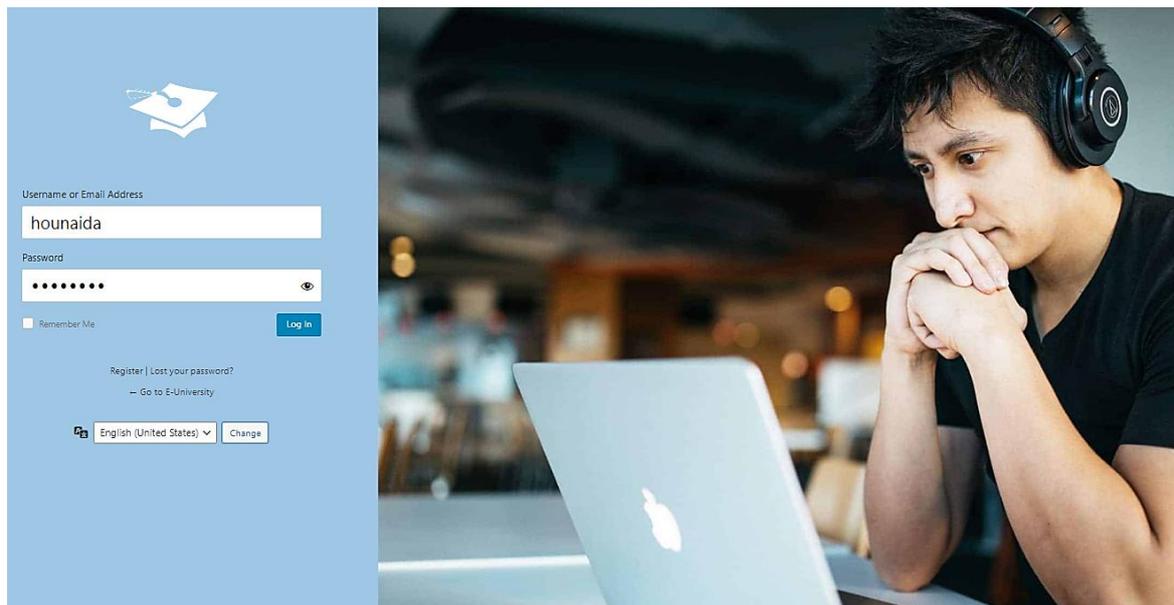


Figure 27 : Interface d'authentification

- **Interface d'accueil**

Si les données entrées par l'utilisateur dans l'étape précédentes ont correctes, une interface s'ouvre, qui permette à lui d'accéder aux différentes fonctionnalités (All courses , ...) fournies par la plateforme. (**Voir figure 28**)



Figure 28 : Interface d'accueil

- **Interface de recherche**

Cette interface (Figure ci-dessous), qui contient tous les cours qui existent dans notre plateforme, Ainsi, à partir de cette interface, l'utilisateur peut sélectionner le cours qui le convient.

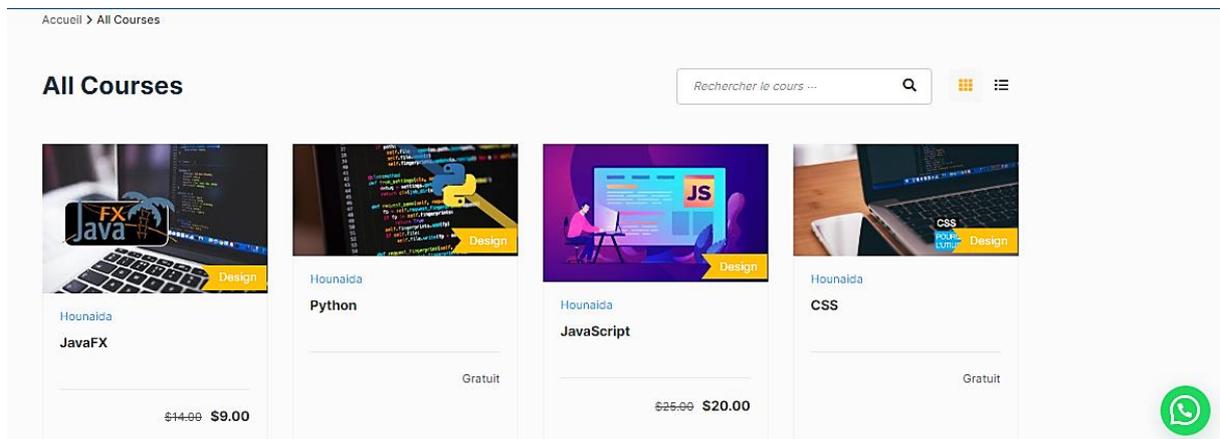


Figure 29: Interface de recherche.

- **Interface de cours :**

Depuis l'interface de recherche (**figure 29**) on peut accéder au page de cours (**figure 30**), dans cette page on peut avoir les informations générales sur le cours comme : titre, nombre d'étudiants inscrits dans ce cours, langue, prix, nombre de leçons et l'enseignant formateur.

L'étudiant doit s'inscrire dans le cours pour accéder au contenu et donner un avis sur ce cours.

Dans cette interface on a le **service de cours (Vidéo, Image, Texte)**, où l'apprenant peut étudier ce cours avec Vidéo, Image ou Texte.

CHAPITRE 5 : Implémentation de la plateforme et test

Et on a aussi dans cette page, le service d'avis de cours (Reviews) (figure 30)

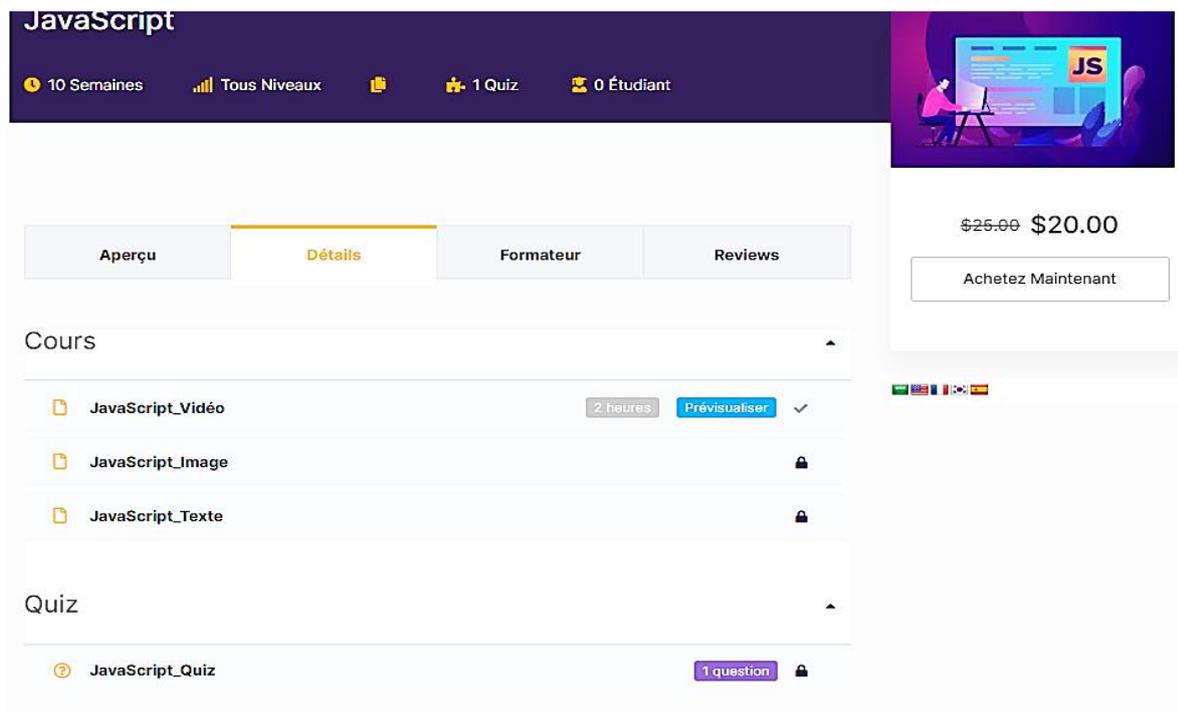


Figure 30 : Interface de cours

- Interfaces des avis de cours (Reviews) :

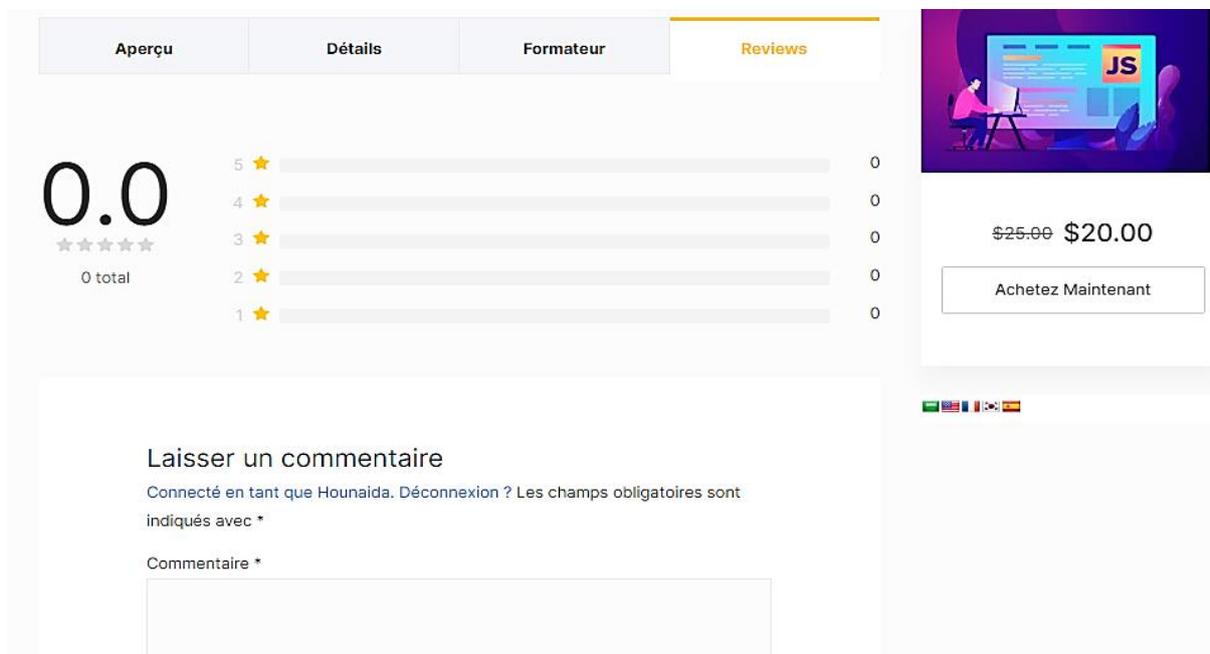


Figure 31 : les avis de cours

- Le service traduction :

Ce service permet à l'utilisateur de choisir la langue qui lui convient (ex : Anglais), pour suivre sa cour.

La figure suivante montre quand ce service être activé ou désactivé :



Traduction désactivé



Traduction activé

Figure 32 : service traduction

- **Le service de communication :**

L'enseignant et l'apprenant ont besoin de communiquer entre eux à l'aide des outils de messageries (Gmail, Facebook, ...) comme le montre cette figure, afin de rendre le cours plus claire et fluide.



Figure 33 : service de communication

- **Le service de désigne :**

Avec cette fonctionnalité, on peut voir 3 couleurs comme types de désigne (Basique, Préférable, Meilleur) comme suivant :

- La couleur noir représente le basique désigne .
- La couleur violet représente le préférable désigne .
- La couleur bleu représente le meilleur désigne .

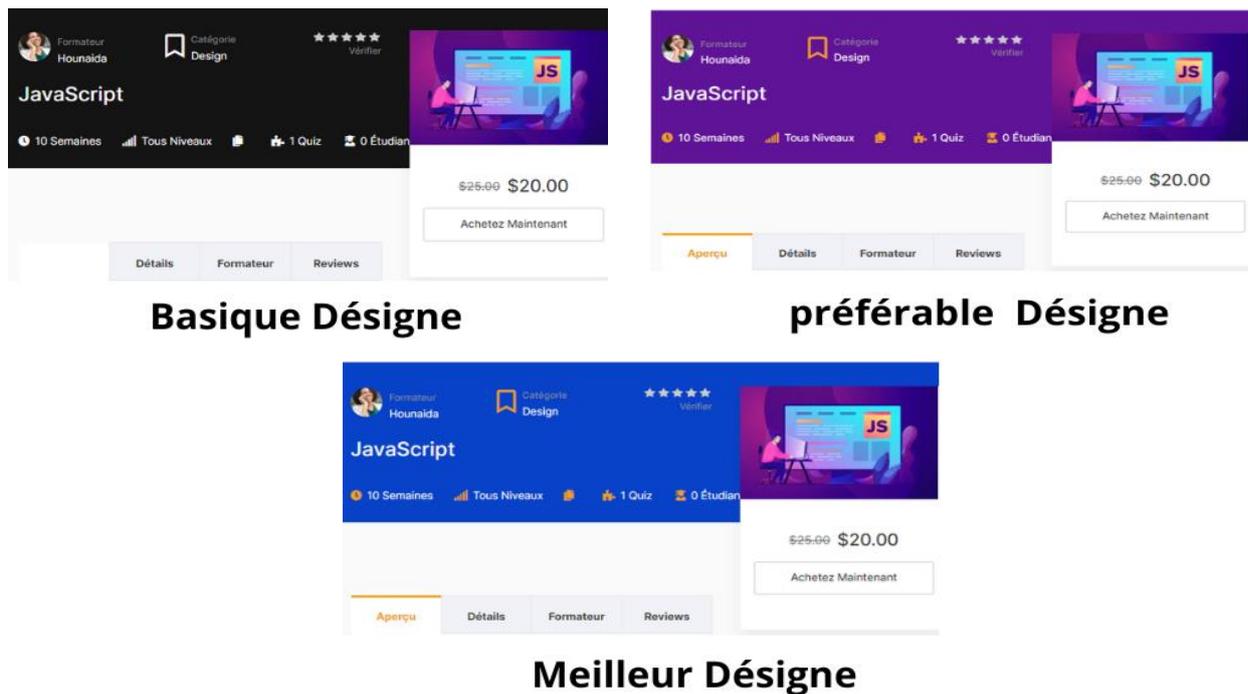


Figure 34 : services désigne

Nous avons montré les interfaces de base, ainsi que les services que nous avons incorporés dans notre plateforme :

- Cours avec 3 formats (Vidéo, Image, Texte).
- Avis de cours.
- Traduction.
- Désigne.
- Message.
- Statistique.

Par conséquent, lorsque l'approche est déclenchée, elle va :

- Modifier la qualité de services spécifiques (Cours, Désigne, Message).
- D'autres services peuvent être désactivés ou activés (Statistiques, Avis de cours, Traduction).

4 Présentation de la simulation du l'approche

La simulation est un test que l'utilisateur effectue pour déterminer l'authenticité du programme en recevant des graphiques comme résultats. Les paramètres de l'algorithme, la simulation et les graphes sont trois composantes essentielles de la simulation de notre approche.

- **L'interface d'accueil :**

Avec cette interface on peut voir les 2 simulations (Avec ou Sans PSO) (voir **figure 35**)



Figure 35 : Interface d'accueil de simulation

- **L'interface de paramétrés d'algorithme PSO :**

Pour analyser le comportement de notre algorithme et la convergence de la fonction de fitness, nous devons passer par l'étape de configuration et sélectionner les paramètres de simulation PSO. Notre application prend en charge la configuration en utilisant l'interface PSO décrite dans **la figure 36**.



Figure 36 : Interface de paramétrés d'algorithme PSO

- **L'interface de paramètres sans d'algorithme PSO :**

Cette interface représente la configuration sans PSO, contient le bouton **Tester** qui permet de générer aléatoirement les particules de population et choisir une particule parmi eux. (Voir figure 37)



Figure 37 : Interface de paramètre sans PSO

5 Expérimentations et résultats

Cette section résume les résultats des tests expérimentaux que nous avons menés sur le cas d'étude E-Learning en utilisant les deux approches alternatives : sans adaptation et d'autre basée sur l'algorithme PSO. Notre application permet de générer des graphiques à l'aide de la bibliothèque **Highcharts**. Ces graphiques montrent les caractéristiques et les résultats d'une ou plusieurs simulations.

Les tests ont été réalisés à l'aide d'un ordinateur portable Acer fonctionnant sous Windows 10 professionnel et équipé d'un processeur Intel(R) Core (TM) i3-3217U CPU @ 1.80GHz 1.80 GHz et de 4.00 GB de RAM.

Nous essaierons d'évaluer notre stratégie par rapport aux critères que nous avons spécifiés précédemment (temps de réponse, utilité et utilisation de la connexion) et d'évaluer l'efficacité de l'utilisation de la plate-forme avec et sans notre approche dans les tests qui suivent :

- Les paramètres d'algorithme PSO.
- La somme de la fonction de fitness avec et sans notre approche.
- L'évolution de fonction fitness dans notre algorithme.
- L'utilité globales des services activés dans la plateforme avec et sans notre approche.
- Temps de réponse de notre de plateforme avec et sans notre approche.
- Consommation des données de connexion avec et sans notre approche.
- Le temps moyen d'exécution de notre méthodologie.

6 Exemple de travail

6.1 Simulation avec PSO :

Les paramètres utilisés dans la simulation pour notre algorithme, sont :

Tableau 20: paramètre de simulation avec PSO

Paramètre	Valeur
Population	10
Paramètre cognitive C1	0.5
Paramètre cognitive C2	0.5

Nombre d'itérations	100
---------------------	-----

6.2 Justification des paramètres :

Dans cette section on va essayer de justifier le choix de paramètre de notre algorithme PSO par rapport au :

- ✓ Les poids des critères qu'on les a déterminés dans les sections précédentes (temps de réponse, utilité, consommation).
- ✓ La valeur de coefficient d'inertie ω , on a choisi la valeur 1 (car $0.8 < \omega < 1.2$)
- ✓ Le temps de réponse maximal est 4000 ms (selon des experts du web).
- ✓ On a choisi aléatoirement la valeur 0.5 pour les deux paramètres cognitifs C1 et C2, car l'intervalle de ces paramètres est entre [0 ,2].
- ✓ Et pour R1 et R2, on a mis dans le code de notre algorithme une fonction qui permet de choisir des valeurs aléatoires dans l'intervalle [0,1].

Maintenant on va essayer de justifier le choix de nombre de population et le nombre d'itération :

a. Nombre population

Ce tableau détermine des résultats de tests sur différents nombres de population pour voir la somme de valeur de fitness, et le temps d'exécution de chaque une, pour vérifier que le nombre de population (10) est bien défini.

Tableau 21 : Test de nombre d'essaim

Taille population	Somme fitness	Temps d'exécution (ms)
5	39,89	95.30
10	77,98	157.99
15	122,11	325.88
20	163,40	673
50	399,41	1049
100	804,62	1874

CHAPITRE 5 : Implémentation de la plateforme et test

On peut voir que le temps d'exécution de l'approche augment avec l'augmentation de nombre de l'essaim. Par exemple lorsque on a un essaim de taille 5, le temps d'exécution est environ 95ms est moins d'un second. Mais avec la taille 100, le temps d'exécution est 1874 ms qui environ de 2seconds. Et tant que pour nous le temps d'exécution est plus important, donc il faut le minimiser le maximum. C'est pour ça on ne peut pas choisir des nombres des essais supérieures à 10 pour garder notre but de l'approche, et afin de se diversifier dans les plans de notre plateforme et pour permettre à l'algorithme de choisir le meilleur plan qui satisfait notre objectif (maximiser l'utilité, minimiser le temps de réponse et la consommation) on ne peut pas mettre le nombre d'essais en dessous de 10.

D'après ça, le nombre d'essaim 10 est bien défini.

b. Nombre itération

Dans cette partie on va vérifier que le paramètre de nombre d'itérations de l'algorithme PSO est bien défini (dans ce graphe on à prendre que les 20 premières itérations).

La figure au-dessous représente l'évolution de fonction de fitness par rapport au nombre d'itération.

La fonction de fitness dans l'algorithme PSO change après l'exécution de la première itération. C'est-à-dire que dans l'itération 0, les particules ont des valeurs appelées valeur de fitness et dans chaque itération les particules prendre une valeur appelée **Gbest** qui est la meilleure valeur de l'essaim.

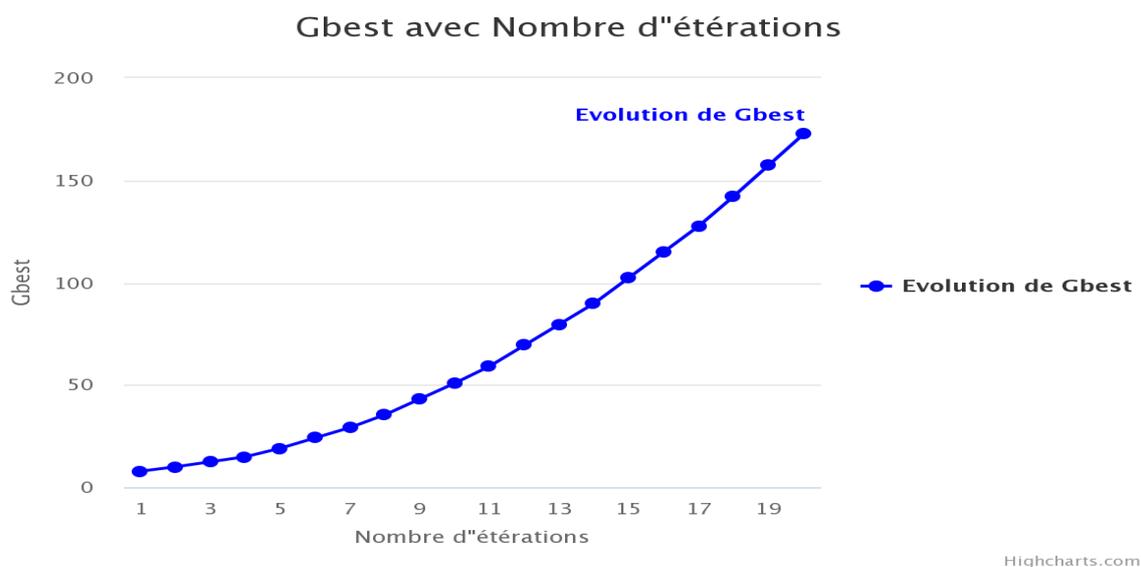


Figure 38 : évolution de Gbest

CHAPITRE 5 : Implémentation de la plateforme et test

Ce test montre que le nombre d'itération approprié pour notre approche est 100, car si on met moins de 100 itérations, notre approche n'arrive pas à la meilleure valeur de **Gbest** c'est-à-dire qu'elle n'arrive pas au meilleur résultat, et d'après le journal de simulation qui nous verrons dans les prochaines sections son remarque que l'algorithme PSO donne un plan acceptable lorsque l'itération est 100, donc le nombre d'itération 100 est bien défini.

6.3 Etat d'essaim initial :

Afin de composer notre population initiale, on a utilisé une fonction qui permet de gérer aléatoirement les variations des services de notre plateforme. Pour illustrer ça, on a fait un exemple de 10 particules comme nombre de population (10 états de plateforme).

Après l'exécution de notre code, et avec un clic sur le bouton **lancer l'algorithme PSO**, une population de 10 particules est initialisée. Comme représenté par le tableau (**tableau 23**) et la figure suivante (**Figure39**) :

Tableau 22 : état initial d'essaim

Code particule	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11
P1	1	0	1	1	3	1	0	0	1	0	2
P2	0	1	2	1	2	1	1	0	0	1	2
P3	2	2	1	0	1	0	0	0	0	0	0
P4	0	1	0	1	1	0	1	2	0	1	2
P5	0	0	0	0	0	1	0	3	1	0	1
P6	1	1	2	0	0	0	0	3	1	0	2
P7	1	1	0	0	1	1	0	0	0	1	1
P8	2	1	0	0	2	1	1	1	1	1	1
P9	2	0	2	0	2	0	0	1	0	0	1
P10	0	0	1	1	3	0	1	0	1	1	1

Voici le journal de simulation :

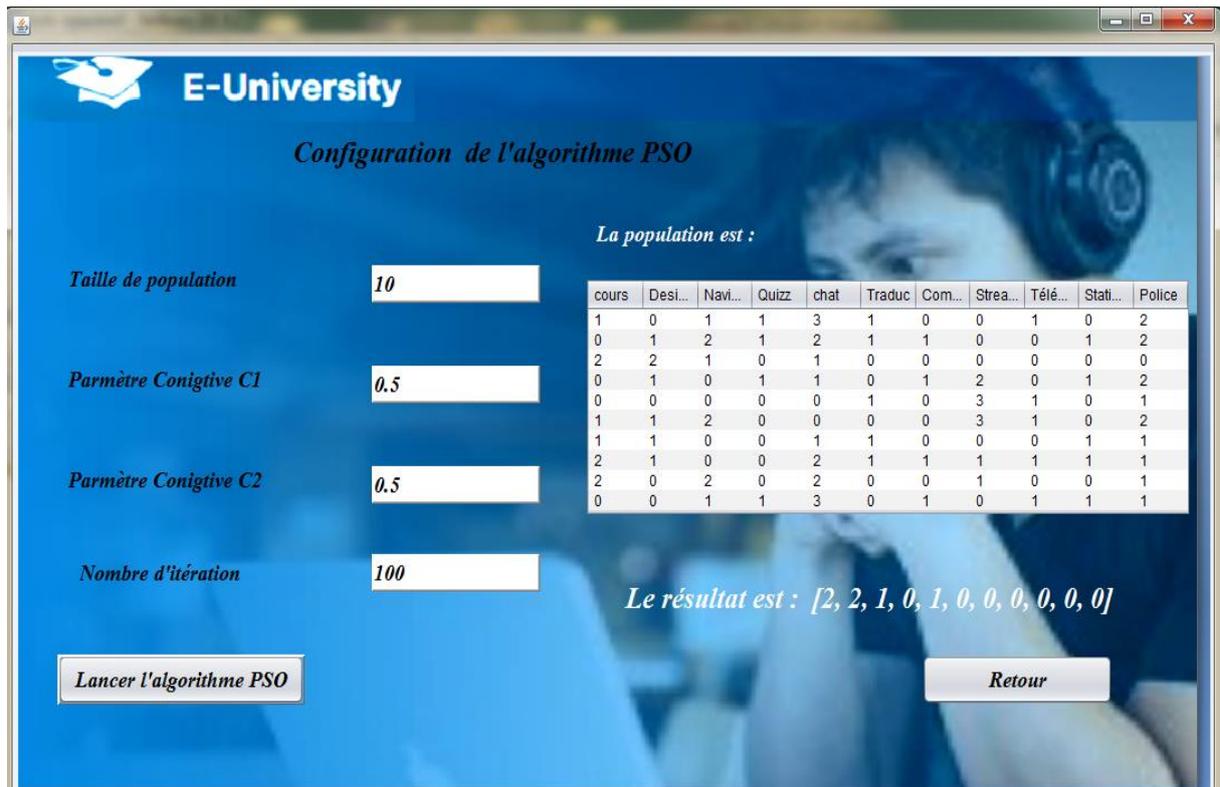


Figure 39 : journal de simulation avec PSO

Le tableau présenté dans l'interface de configuration de l'algorithme PSO, représente l'ensemble des particules avec leurs codes de variations (étudier précédemment). Par exemple les comportements de code de particule 1 (**la ligne 1 de tableau**) sont :

Particule 1 : {Cours : image, Design : meilleur, Navigation Liens : méta, Quizz : Activé, Message : Texte, Traduction : Activé, Commentaires : Désactivé, Streaming vidéo (Qualité) : Désactivé, Téléchargement : activé, Statistique : Désactivé, Police : Petite}

Après le lancement de notre algorithme PSO, l'algorithme choisit la particule 3 après 100 itérations :

Particule 3 : {Cours : Texte, Design : Basique, Navigation Liens : Méta, Quizz : Désactivé, Message : Vidéo, Traduction : Désactivé, Commentaires : Désactivé, Streaming vidéo (Qualité) : Désactivé, Téléchargement : Désactivé, Statistique : Désactivé, Police : Désactivé}

6.4 Simulation sans PSO :

Pour voir le résultat de simulation sans PSO, on a fait un test avec le paramètre suivant :

Tableau 23 : paramètre de simulation sans PSO.

Taille population 10

Voici le journal du test :



Figure 40 : journal de simulation sans PSO

Après le clic sur le bouton **Tester**, une population de 10 particules est générée, et choisir la particule 5 comme résultat, comme le montre dans (la Figure40).

Particule 5 : {Cours : Vidéo, Design : Meilleur, Navigation Liens : Tous, Quiz : Activé, Message : Audio, Traduction : Activé, Commentaires : Activé, Streaming vidéo (Qualité) : Faible, Téléchargement : Activé, Statistique : Activé, Police : Petite}.

7 Temps de réponse

Le critère le plus essentiel dans notre méthode est le temps de réponse, nous avons établi un temps limite de réponse (4000 ms) que notre plateforme ne doit pas le dépasser avec notre approche. Nous évaluerons le temps de réponse de la plateforme par rapport à nombre d'itération de l'algorithme.

La figure ci-dessous montre le graphe de temps de réponse de notre plateforme avec l'algorithme PSO par rapport aux itérations de l'algorithme.

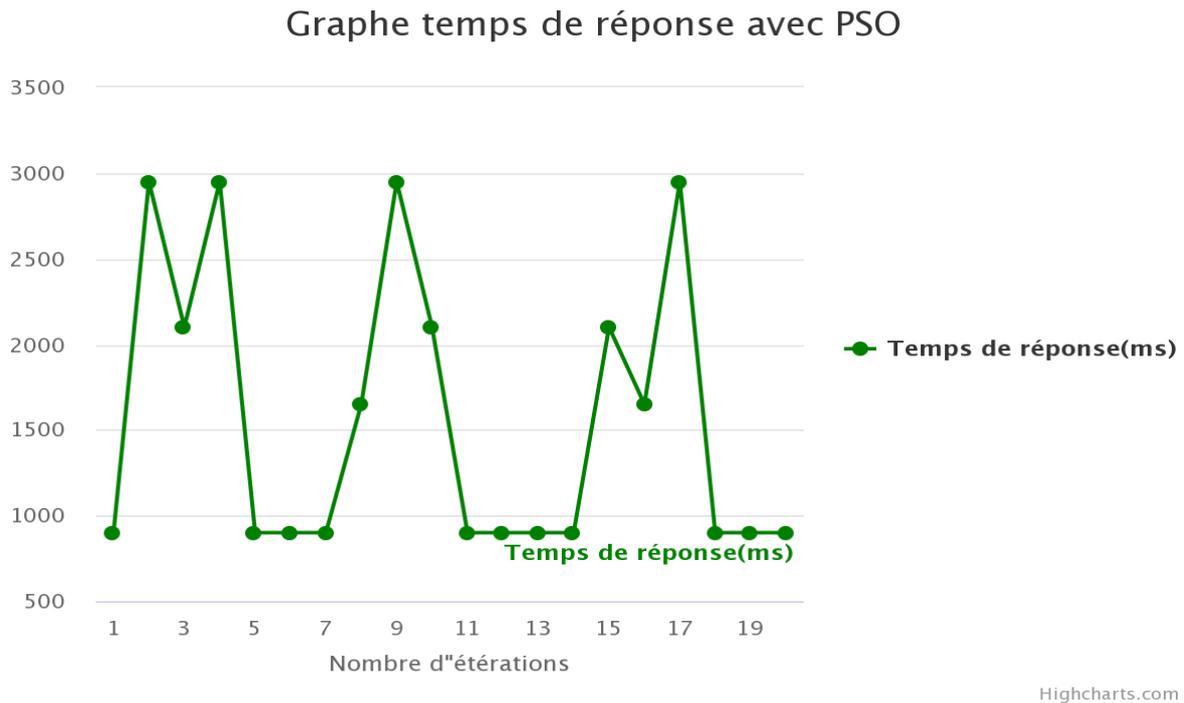


Figure 41 : Teste de temps de réponse

D'après la figure, on remarque que l'algorithme PSO à chaque itération, il essaie de nous donner des valeurs satisfaisantes concernant le temps de réponse de notre plateforme dans l'intervalle [900ms, 3000ms] qui ne nous dépasse pas notre seuil (4000ms), comme nous l'avons dit précédemment (ne doit pas dépasser ce seuil), on peut voir que les dernières itérations nous donnent une solution raisonnable, c'est le meilleur temps de réponse pour un utilisateur dans notre plateforme.

Alors on peut conclure que les résultats obtenus sont satisfaisants, où le temps de réponse est raisonnable et bien optimisé. Ce qui signifie que notre approche a bien aidé la plateforme de s'adapter aux faibles vitesses de connexion d'utilisateur.

8 Consommation de connexion

L'objectif de notre approche est de diminué la consommation des données de connexion, pour cela on a essayé de faire des tests sur notre approche et on a généré des graphes pour noter les changements des valeurs de consommation par rapport au nombre d'itération de l'algorithme.

La figure au-dessous représente le graphe de consommation de données de connexion (Kbps) avec notre approche par rapport au nombre d'itération de l'algorithme.

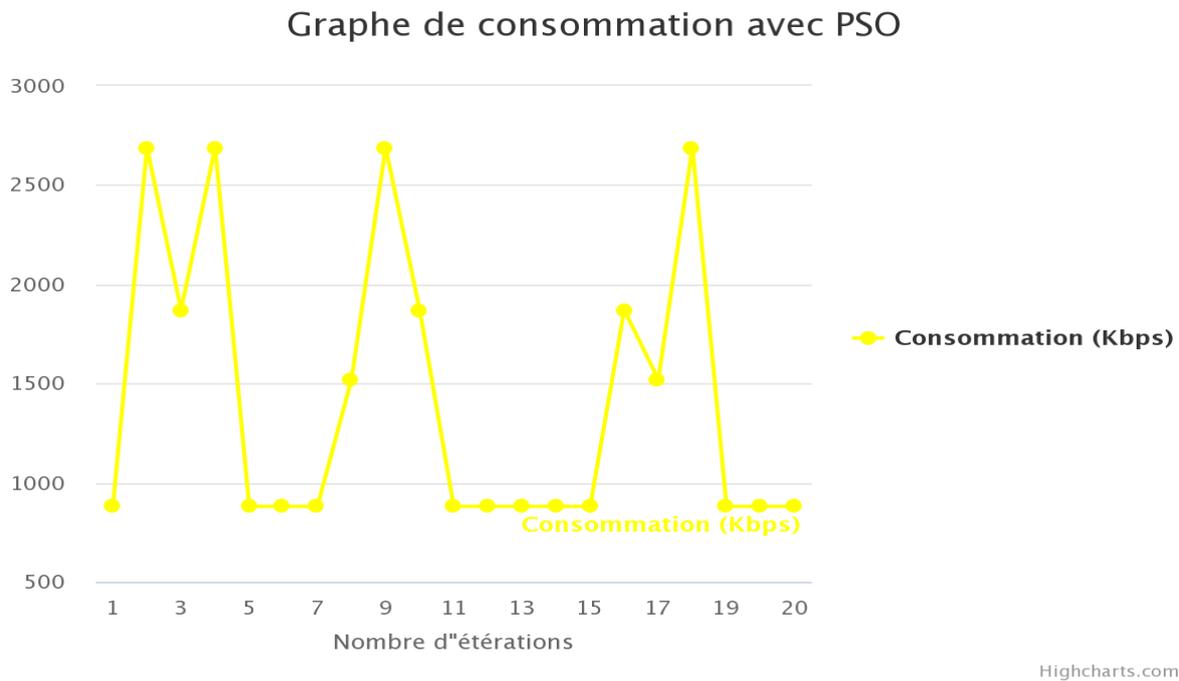


Figure 42 : Test 1 de consommation des connexions

Les valeurs de consommation avec notre approche sont changées à chaque itération dans l'intervalle [900kbps, 2700kbps]. C'est des valeurs raisonnables, à la fin de notre l'algorithme les valeurs de consommation restent stables environ (1000kbps) de numéro d'itération 20 ou plus, cela représente les bons résultats et c'est notre but.

Enfin, nous concluons que notre algorithme a réussi d'atteindre notre objectif, de réduire la consommation de données connexion dans les faibles vitesses de connexion d'utilisateur.

9 Utilité :

Le troisième objectif de notre approche est de maximiser l'utilité des services. On a testé l'utilité des services activés dans la plateforme avec notre approche par rapport au nombre d'itérations de l'algorithme.

La figure suivante représente le graphe d'utilité des services de notre plateforme avec notre approche par rapport au nombre d'itérations de l'algorithme.

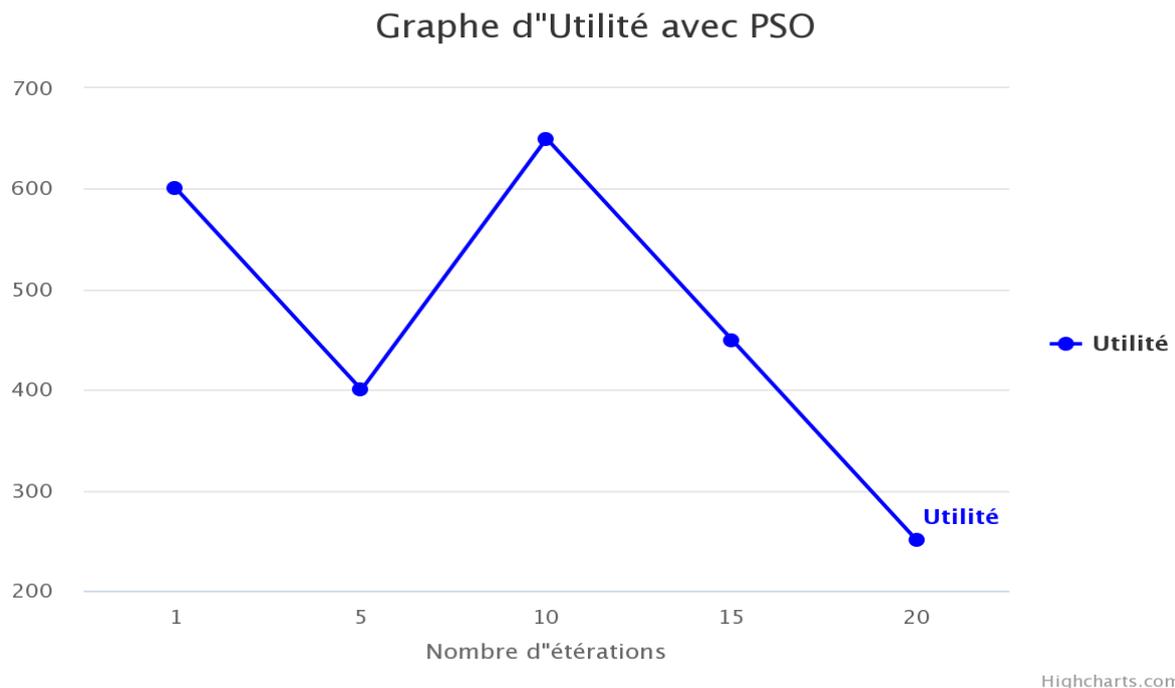


Figure 43 : Test d'utilité

Avec notre approche (avec PSO) on voit que les valeurs d'utilité des services a diminuées, d'autre part, nous savons que plus utilité des services a augmenté plus le temps de réponse de plateforme est augmenté, donc on peut dire que notre approche essaie de garder des valeurs raisonnables concerne l'utilité et maintient d'autre part sur les valeurs temps de réponse de plateforme car le temps de réponse est plus important que l'utilité (le poids de temps de réponse (0.6) qu'on a fixé précédemment et plus grand que le poids d'utilité (0.2) .

Sur la base de graphe que nous avons fait on peut conclure que notre approche offre des solutions raisonnables concernant les valeurs d'utilité en considérant les valeurs de temps de réponse de la plateforme.

10 Temps d'exécution d'approche :

Après plusieurs exécutions de notre algorithme et voir le temps d'exécution de chaque une. Le temps d'exécution de l'algorithme d'Essaime particulaire est compris entre 152ms et 384ms (moyen de temps d'exécution est 286ms) par une simulation, c'est-à-dire moins d'une seconde, ce qui est très raisonnable et acceptable.

11 Comparaison entre l'algorithme génétique (GA) et PSO :

Dans cette partie on a fait une comparaison entre les deux algorithmes d'optimisation PSO et GA, l'algorithme GA l'un des algorithmes et des méthodes d'optimisation (les algorithmes de méta heuristique), qui a utilisé dans le même cas que nous[7]. Les figures au-dessous représentent les changements des valeurs de temps de réponse, consommation de donnée et utilité par rapport au temps (ms) :

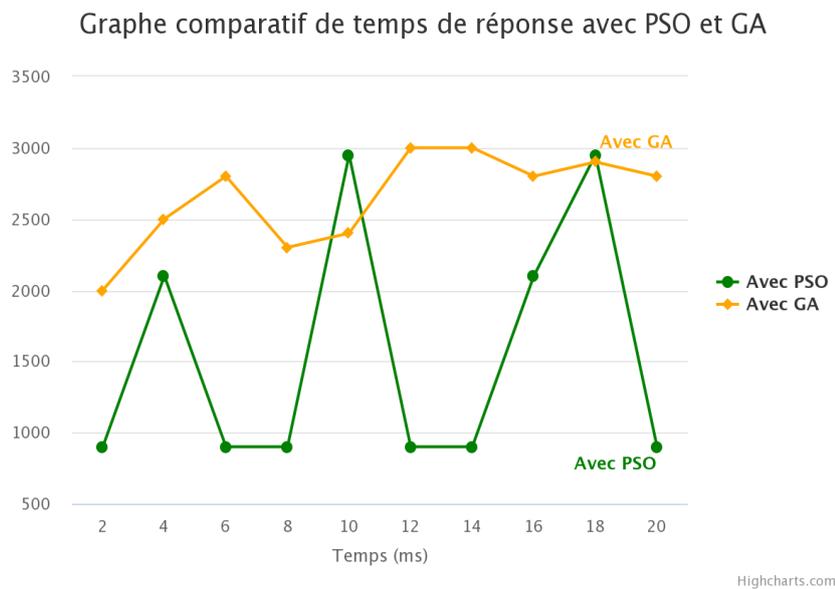


Figure 44 : comparaison du temps de réponse entre GA et PSO

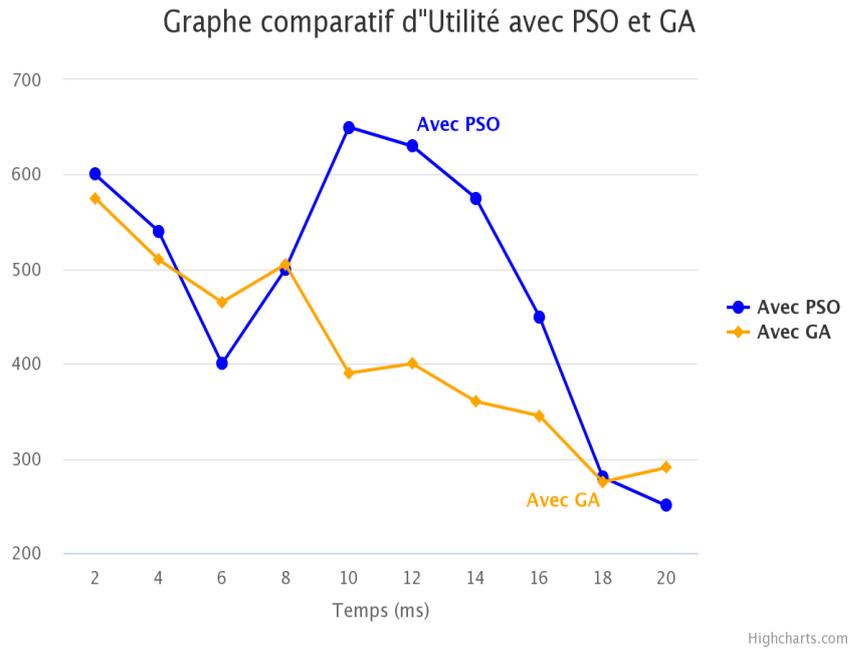


Figure 45: comparaison d'utilité entre GA et PSO

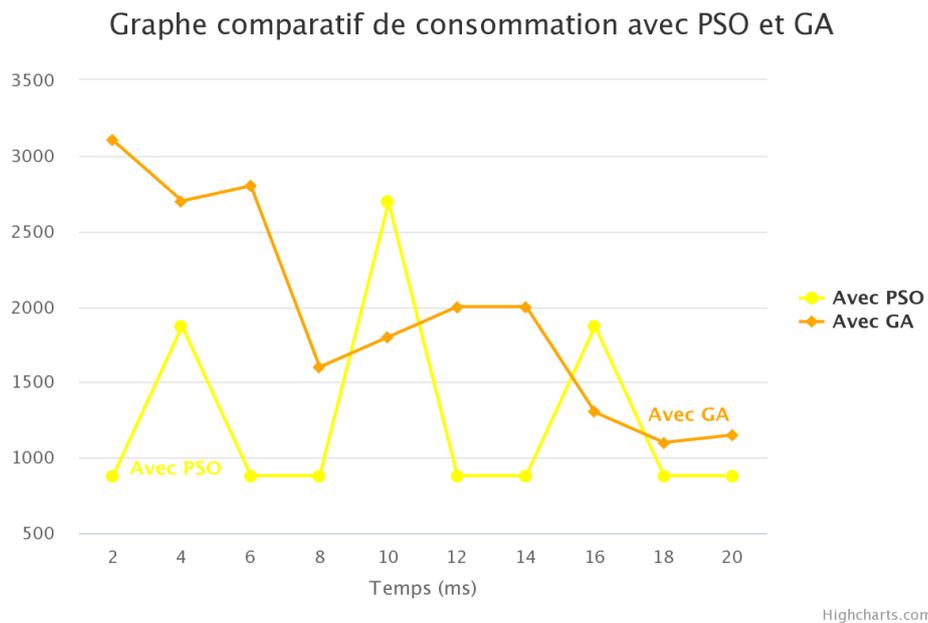


Figure 46: comparaison de consommation entre GA et PSO.

12 Analyse des graphes :

- **Comparaison de temps de réponse entre GA et PSO :** on remarque que les valeurs de temps de réponse changent avec le temps (ms). Avec GA les valeurs changent dans l'intervalle [2000, 3000[et l'algorithme converge vers la valeur 2900 au moment 20 ms. Et concernant l'algorithme PSO les valeurs de temps de réponse

CHAPITRE 5 : Implémentation de la plateforme et test

changent dans l'intervalle [900 ,3000] c'est-à-dire ne dépasse pas notre seille de l'approche (4000ms) et l'algorithme converge vers la valeur 900 au moment 20ms.

- **Comparaison d'Utilité entre GA et PSO** : les valeurs d'utilité des services activée de plateforme diminuent avec le temps, c'est pour l'algorithme GA dans lequel la valeur maximale est 590 et la valeur minimale est 300. Cependant les valeurs d'utilité concernant l'algorithme PSO changent dans l'intervalle [250 ,650] ,650 valeurs maximales 250 valeurs minimales où il converge l'algorithme PSO.
- **Comparaison de Consommation de donnée entre GA et PSO** : pour l'algorithme GA la consommation diminue avec le temps] 3000, 1200] et pour PSO les valeurs changent dans [900,2700].

13 Discussion :

Après les analyses des graphes qu'on a faits précédemment, on peut déduire que les deux algorithmes sont bien optimisés les trois critères, c'est-à-dire qu'ils ont satisfaire notre problème d'optimisation (minimisation du temps de réponse et la consommation de données avec maximisation de l'utilité des fonctionnalités) et donnent un bon résultat lorsque les services de la plateforme sont activés, cependant l'algorithme PSO est meilleur que l'algorithme GA. La deuxième différence notable concerne le temps d'exécution de l'algorithme, l'algorithme GA besoin de (70ms) pour trouver le bon plan d'adaptation, par contre l'algorithme PSO besoin de (268 ms), cette différence est due à cause de la configuration les deux algorithmes (les opérations, les instructions ...), donc dans ce cas GA meilleure que PSO.

14 Conclusion :

Dans ce dernier chapitre, nous avons présenté notre plateforme E-Learning avec les différents outils utilisées pour le développement. Ainsi que l'ensemble des tests appliqués et les résultats obtenus, qu'on a présentés par des graphes explicatifs.

Conclusion générale

Dans ce mémoire, nous avons présentée la nouvelle technologie « l'auto-adaptation », qui est utilisée par les grandes entreprises dans le monde dans différents domaines, qui a contribué à résoudre la plupart des problèmes notamment le contexte d'exécution des systèmes modernes s'exécutent change continuellement, donc l'application de ces systèmes nécessitent une prise en charge pour l'auto-adaptation aux changements continus.

Dans le cadre de notre étude, Nous avons présenté notre approche qui a consisté à la conception et le développement d'une plateforme E-Learning dotée de la fonctionnalité d'auto adaptation.

En effet, l'approche suggérée permet d'adapter l'utilisation des différents services de la plateforme E-Learning à un moment donné en fonction de la vitesse de connexion de l'utilisateur. Pour ce faire l'approche est construite sur une technique d'optimisation basée sur des métaheuristiques (algorithme PSO).

Nous avons commencé notre travail par une étude bibliographique sur les systèmes auto-adaptatifs. Ensuite, nous avons fait une analyse métaheuristique basée principalement sur PSO que nous avons utilisé dans notre approche.

Par la suite, nous avons montré la conception et l'implémentation de l'approche étudiée, ainsi que les solutions d'auto-adaptation recommandées basées sur l'algorithme PSO.

Et à la fin, nous avons montré les tests et validations de l'approche proposée. Et d'après nos résultats, on conclut que l'algorithme PSO a bien satisfait notre objectif.

Comme perspectives, notre futur objectif est de :

1. poursuivre le développement de notre approche d'adaptation des systèmes en utilisant d'autres algorithmes d'optimisation.
2. Généraliser notre approche à d'autres problèmes plus complexes dans d'autres domaines.

Enfin, ce travail nous a été très enrichissant. Il nous a permis de toucher de près une partie très importante dans le développement des systèmes auto-adaptatif.

Bibliographie:

- [1] D. Weyns, *An introduction to self-adaptive systems : a contemporary software engineering perspective*. 2021.
- [2] Y. Ayoub *et al.*, “Patrons temporels pour spécifier les systèmes auto-adaptatifs To cite this version : HAL Id : hal-02437249 Patrons temporels pour spécifier les systèmes auto-adaptatifs,” 2020.
- [3] P. Oreizy *et al.*, “An architecture-based approach to self-adaptive software,” *IEEE Intell. Syst. Their Appl.*, vol. 14, no. 3, pp. 54–62, 1999, doi: 10.1109/5254.769885.
- [4] L. T. Mazeiar Salehie, “Towards a goal-driven approach to action selection in self-adaptive software,” *Softw. - Pract. Exp.*, vol. 39, no. 7, pp. 701–736, 2009, doi: 10.1002/spe.
- [5] M. Camus, “Système auto-adaptatif générique pour le contrôle de robots ou d’entités logicielles,” no. September 2007, 2007, doi: 10.13140/RG.2.1.1688.3605.
- [6] B. Khalil and A. Dahmane, “Optimisation de la planification de l’auto-adaptation des systèmes par l’utilisation des métaheuristiques,” 2019.
- [7] B. Acil and A. Nabil, “Vers un système auto-adaptatif pour le E-learning Présenté,” 2011.
- [8] F. Chauvel, “Méthodes et outils pour la conception de systèmes logiciels auto-adaptatifs,” no. September 2008, 2014.
- [9] G. De Pietro, L. Gallo, R. J. Howlett, and L. C. Jain, “Intelligent interactive multimedia systems and services 2017,” *Intell. Interact. Multimed. Syst. Serv. 2017*, no. May, 2018, doi: 10.1007/978-3-319-59480-4.
- [10] M. L. BERKANE, “Intégration des technologies « Patrons de conception » et « Aspect » dans les applications autonomiques,” 2012.
- [11] R. El Ballouli, “Modeling self-configuration in Architecture-based self-adaptive systems Rim El Ballouli To cite this version : HAL Id : tel-02175324 Modeling Self-configuration in Architecture-based Mod elisation de la configuration automatique dans,” no. July, 2019, doi: 10.13140/RG.2.2.19830.55366.
- [12] R. De Lemos *et al.*, “Software engineering for self-adaptive systems: A second

Bibliographie

- research roadmap,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 7475 LNCS, no. June, pp. 1–32, 2013, doi: 10.1007/978-3-642-35813-5_1.
- [13] B. KARIMA, “Conception d’un système multi -agents adaptatif pour la résolution de problème,” 2016.
- [14] “Un retour et une reconnaissance pour l’ASTROLab à Distance.” <https://www.msn.com/fr-ca/actualites/Région de Sherbrooke/un-retour-et-une-reconnaissance-pour-lastrolab-à-distance/ar-AAKUkfu> (accessed Feb. 26, 2022).
- [15] A. BOULEDROUA, “Découverte des préférences des apprenants dans un environnement de e-learning,” 2012.
- [16] A. J. Romiszowski, “How’s the e-learning baby? Factors leading to success or failure of an educational technology innovation,” *Educ. Technol.*, vol. 44, no. 1, pp. 5–27, 2004.
- [17] E. L. Y. Youssef, “Du e-learning au digital learning : enjeux et perspectives à l ’ ère de la transformation digitale From e-learning to digital learning : issues and perspectives in the era of digital transformation,” vol. 5, no. 2022, pp. 164–180.
- [18] S. AZOUGH, “E-Learning Adaptatif: Gestion intelligente des ressources pédagogiques et adaptation de la formation au profil de l’apprenant,” vol. 212, no. 0, 2014.
- [19] A. Qazdar and C. Cherkaoui, “Vers des Systèmes E-learning Adaptatifs : une étude des capacités d ’ adaptation,” no. October, 2015, doi: 10.13140/RG.2.1.5132.6163.
- [20] P. Brusilovsky and P. M. E. De Bra, “Adaptive hypertext and hypermedia: proceedings of the 2nd workshop, Pittsburgh, Pa., June 20-24, 1998,” vol. 9812, no. 1998, 1998.
- [21] “ELM-ART: Registration.” <http://art2.ph-freiburg.de/art/login-e.html> (accessed Mar. 25, 2022).
- [22] C. Wolf, “iWeaver : Towards ’ Learning Style ’ -based e-Learning in Computer Science Education,” *Fifth Australas. Comput. Educ. Conf. (ACE 2003)*, 2003. *Aust. Comput. Soc. 2003 CRPIT ISBN 0-909925-98-4, p273-279*, vol. 20, pp. 273–279, 2003.
- [23] D. Hauger and M. Köck, “State of the art of adaptivity in e-learning platforms,” *LWA*

Bibliographie

- 2007 - *Lernen - Wissen - Adapt. - Learn. Knowledge, Adapt. Work. Proc.*, pp. 355–360, 2007.
- [24] A. El Dor, “Perfectionnement des algorithmes d’optimisation par essaim particulaire. Applications en segmentation d’images et en électronique,” p. 148, 2012.
- [25] M. A. C. M. Bensmail, “Une méta-heuristique hybride pour une meilleure localisation des nœuds dans les RCSFs .,” 2021.
- [26] B. Mille, “Méthodes approchées pour la résolution d ’ un problème d ’ ordonnancement avec travaux interférants,” vol. 33, no. 0, 2014.
- [27] N. KHERICI, “Méthodes de résolution en optimisation combinatoire,” vol. 78, no. c, pp. 8–10, 2013.
- [28] M. D. E. A. A. Berrim Ali, “Prédiction de la tension de contournement par l’approche d’optimisation PSO,” pp. 1–66, 2013.
- [29] “des nuées d’oiseaux research gat – Recherche Google.”
<https://www.google.com/search> (accessed Jul. 29, 2022).
- [30] “Banc de poissons : et si l’Homme s’inspirait de leur esprit collectif? Fishipedia.”
<https://www.fishipedia.fr/article/banc-de-poissons-et-si-lhomme-sinspirait-de-leur-esprit-collectif/> (accessed Jul. 29, 2022).
- [31] N. Lazaar, “Optimisation des alimentations électriques des Data Centers,” no. April, 2021.
- [32] B. L. M. Amine, “Optimisation par essaim particulaire et par algorithme génétique pour la gestion intelligente des puissances d’un micro-réseau,” 2020.
- [33] H. OUAFI, “Contribution à l’optimisation des réseaux électriques en présence des Multi-FACTS par des méthodes métaheuristiques hybrides,” no. December, 2019.
- [34] N. Di Cesare, “Développement d ’ une nouvelle méthode metaheuristique pour l ’ optimisation topologique des structures et des metamatériaux,” no. November 2016, 2017.
- [35] Chaker Amine, “Clustering de données dans un contexte Big Data,” 2019.
- [36] A. Ben Khalifa, “Contribution aux techniques de fusion des modalités biométriques,” no. March 2014, 2018, doi: 10.13140/RG.2.2.11432.26881.

Bibliographie

- [37] E. Bagheri, F. Ensan, D. GASEVIC, and M. Boškovic, “Modular feature models: Representation and configuration,” *J. Res. Pract. Inf. Technol.*, vol. 43, no. 2, pp. 109–140, 2011.
- [38] “Diagramme De Cas d’utilisation.”
- [39] C. S. DIALLO, “CONCEPTION ET IMPLEMENTATION D’UNE APPLICATION DE GESTION SCOLAIRE DE L’INSTITUT AFRICAIN TRADING BOURSIER,” pp. 2021–2022.
- [40] R. Madal, “Conception et réalisation d ’ un portail Web. Sciences de l’information et de la com- munication. 2016. dumas-01810600 HAL.”
- [41] “Qu’est-ce qu’un thème WordPress ?” <https://wpmarmite.com/glossaire/theme>. (accessed Aug. 09, 2022).
- [42] “Qu’est-ce qu’une extension dans WordPress ? Glossaire WPMarmite.” <https://wpmarmite.com/glossaire/extension-plugin> (accessed Aug. 09, 2022).
- [43] “NetBeans IDE | Oracle France.” <https://www.oracle.com/fr/tools/technologies/netbeans-ide.html> (accessed Aug. 11, 2022).
- [44] Guillaume Fallet, “Calculateur d’impact pour la transition énergétique,” 2016.