



**HAL**  
open science

# Contribution à l'évaluation de l'apprenant et l'adaptation pédagogique dans les plateformes d'apprentissage : une approche fondée sur les traces

Soraya Chachoua

## ► To cite this version:

Soraya Chachoua. Contribution à l'évaluation de l'apprenant et l'adaptation pédagogique dans les plateformes d'apprentissage : une approche fondée sur les traces. Environnements Informatiques pour l'Apprentissage Humain. Université de La Rochelle, 2019. Français. NNT : 2019LAROS003 . tel-02318386

**HAL Id: tel-02318386**

**<https://theses.hal.science/tel-02318386>**

Submitted on 17 Oct 2019

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



**UNIVERSITÉ DE LA ROCHELLE**

**ÉCOLE DOCTORALE S2IM**

**LABORATOIRE L3I**

**THÈSE** Présentée et soutenue par :

**Soraya CHACHOUA**

Soutenue le : [10 janvier 2019]

Pour obtenir le titre de : **Docteur de l'Université de La Rochelle**

Discipline : **SCIENCES ET INGÉNIERIE POUR L'INFORMATION,  
MATHÉMATIQUES**

**Contribution à l'évaluation de l'apprenant et l'adaptation  
pédagogique dans les plateformes d'apprentissage.**

**Une approche fondée sur les traces**

---

**MEMBRES DU JURY :**

**Rapporteurs :** Ladjel BELLATRECHE, Professeur des universités, Université de Poitiers, LIAS-ENSMA  
Djamal BENSLIMANE, Professeur des universités, Université Claude Bernard Lyon 1, LIRIS

**Examinatrice :** Florence SÈDES, Professeur des universités, Université Toulouse III, IRIT

**Directeur :** Pascal ESTRAILLIER, Professeur des universités, Université de La Rochelle

**Co-directeur :** Jamal MALKI, Maître de conférences, Université de La Rochelle



# Remerciements

Je tiens tout d'abord à remercier mon directeur de thèse, Pascal Estrailier qui m'a accueillie au sein de son agréable équipe de recherche L3i-ADAPT au laboratoire L3I. Tout au long de ces cinq années, j'ai pu apprécier sa vision sur les systèmes interactifs adaptatifs à base de situation, son intégrité, sa disponibilité malgré son calendrier chargé, et sa générosité à partager ses connaissances. Je voudrais lui témoigner ici toute ma gratitude de m'avoir guidée durant ces années et laissée entrevoir ce que le mot recherche veut dire selon son schéma magique « cadrage recherche et développement ».

Je remercie également mon co-encadrant Jamal Malki pour son soutien, sa patience, ses nombreux conseils et l'intérêt qu'il a porté à mon travail durant ma thèse.

Je remercie très sincèrement mon cher ami Nouredine Tamani qui m'a aidée et soutenue dans les moments difficiles de la thèse. J'ai apprécié sa gentillesse, sa collaboration précieuse, sa générosité scientifique et intellectuelle sans oublier le temps qu'il m'a accordé pour m'expliquer de nombreuses notions en mathématiques, logique, logique floue et dans d'autres domaines liés à mes recherches, ainsi que pour son soutien durant la rédaction des articles scientifiques et ses multiples relectures de mon manuscrit de thèse.

Je remercie aussi les membres du jury pour avoir accepté d'examiner ce travail. Je suis reconnaissante pour votre précieux temps que vous avez sacrifié pour ce travail. Je remercie Ladjel Bellatreche et Djamel Benslimane d'avoir accepté de rapporter cette thèse. Je remercie Florence Sèdes d'avoir accepté d'examiner ce travail.

Je voudrais aussi exprimer mon amitié aux personnes avec qui j'ai eu le plaisir de partager ces années de thèse, mon équipe L3i-ADAPT, toute l'équipe administrative du laboratoire L3I et celle de l'école doctorale S2IM et tout.e.s mes ami.e.s du laboratoire L3I.

Je souhaite exprimer toute ma reconnaissance à mes parents, mes sœurs et frères qui m'ont soutenue durant tout mon cursus d'étude et qui m'ont toujours encouragée à faire ce que je souhaite.

Enfin, un grand merci pour toute personne qui a laissé un souvenir dans la mémoire de ce précieux chapitre de ma vie durant mes années de thèse.



## Résumé

L'adoption des Nouvelles Technologies de l'Information et de la Communication (NTIC) a permis la modernisation des méthodes d'enseignement dans les systèmes d'apprentissage en ligne comme l'e-Learning, les systèmes tutoriels intelligents, etc. Ces derniers assurent, une formation à distance qui répond aux besoins des apprenants. Un aspect très important à prendre en considération dans ces systèmes est l'évaluation précoce de l'apprenant en termes d'acquisition des connaissances. En général, trois types d'évaluation et leurs relations sont nécessaires durant le processus d'apprentissage, à savoir : (i) *diagnostic* qui est exécuté avant l'apprentissage pour estimer le niveau des élèves, (ii) *évaluation formative* qui est appliquée lors de l'apprentissage pour tester l'évolution des connaissances et (iii) *évaluation sommative* qui est considérée après l'apprentissage pour évaluer l'acquisition des connaissances. Ces méthodes peuvent être intégrées d'une manière semi-automatique, automatique ou adaptée aux différents contextes de formation, par exemple dans le domaine d'apprentissage des langues (français, anglais, etc.), des sciences fondamentales (mathématiques, physique, chimie, etc.) et langages de programmation (java, python, sql, etc).

Cependant, les méthodes d'évaluation usuelles sont statiques et se basent sur des fonctions linéaires qui ne prennent en considération que la réponse de l'apprenant. Elles ignorent, en effet, d'autres paramètres de son modèle de connaissances qui peuvent divulguer d'autres indicateurs de performance. Par exemple, le temps de résolution d'un problème, le nombre de tentatives, la qualité de la réponse, etc. Ces éléments servent à détecter les traits du profil, le comportement ainsi que les troubles d'apprentissage de l'apprenant. Ces paramètres additionnels sont vus dans nos travaux de recherche comme des traces d'apprentissage produites par l'apprenant durant une situation ou un contexte pédagogique donné.

Dans ce cadre, nous proposons dans cette thèse une approche d'évaluation de l'apprenant à base des traces d'apprentissage qui peut être exploitée dans un système d'adaptation de la ressource et/ou de la situation pédagogique. Pour l'évaluation de l'apprenant, nous avons proposé trois modèles génériques d'évaluation qui prennent en considération la trace temporelle, le nombre de tentatives et leurs combinaisons. Ces modèles ont servi, par la suite, comme métrique de base à notre modèle d'adaptation de la ressource et/ou de la situation d'apprentissage.

Le modèle d'adaptation est également fondé sur les trois traces susmentionnées et sur nos modèles d'évaluation. Notre modèle d'adaptation génère automatiquement des trajectoires d'apprentissage adaptées en utilisant un modèle d'état-transition. Les états présentent des situations d'apprentissage qui consomment des ressources et les transitions entre situations expriment les conditions nécessaires à remplir pour passer d'une situation à une autre. Ces

concepts sont aussi implémentés dans une ontologie du domaine et un algorithme d'adaptation a été également proposé. L'algorithme assure deux types d'adaptation : (i) Adaptation de la situation et (ii) Adaptation des ressources dans une situation.

Afin de collecter les traces d'apprentissage pour la mise en œuvre de notre approche d'évaluation de l'apprenant et d'adaptation de ressources et de situations d'apprentissage, nous avons effectué des expérimentations sur deux groupes d'étudiants en Licence Informatique (L2). Un groupe en apprentissage classique et un groupe en apprentissage adapté. Sur la base des traces obtenues des sessions de travail des étudiants, nous avons utilisé nos modèles d'évaluation dont les résultats ont été utilisés pour mettre en œuvre l'adaptation.

Après comparaison des résultats de l'apprentissage adapté à ceux obtenus de l'apprentissage classique, nous avons constaté une amélioration des résultats en termes de moyenne générale et d'écart-type des moyennes des apprenants.

**Mots-clés:** e-Learning, système tutoriel intelligent (STI), évaluation, adaptation, traces.





## Abstract

The adoption of new Information and Communication Technologies (ICT) has enabled the modernization of teaching methods in online learning systems such as e-Learning, intelligent tutorial systems (ITS), etc.

These systems provide a remote training that which meets the learner needs. A very important aspect to consider in these systems is the early assessment of the learner in terms of knowledge acquisition. In general, three types of assessment and their relationships are needed during the learning process, namely : (i) *diagnostic* which is performed before learning to estimate the level of students, (ii) *formative evaluation* which is applied during learning to test the knowledge evolution and (iii) *summative evaluation* which is considered after learning to evaluate learner's knowledge acquisition.

These methods can be integrated into a semi-automatic, automatic or adapted way in different contexts of formation, for example in the field of languages literary learning such as French, English, etc., hard sciences (mathematics, physics, chemistry, etc.) and programming languages (java, python, sql, etc.). However, the usual evaluation methods are static and are based on linear functions that only take into account the learner's response. They ignore other parameters of their knowledge model that may disclose other performance indicators. For example, the time to solve a problem, the number of attempts, the quality of the response, etc. These elements are used to detect the profile characteristics, behavior and learning disabilities of the learner. These additional parameters are seen in our research as learning traces produced by the learner during a given situation or pedagogical context.

In this context, we propose in this thesis a learner evaluation approach based on learning traces that can be exploited in an adaptation system of the resource and/or the pedagogic situation. For the learner assessment, we have proposed three generic evaluation models that take into consideration the temporal trace, number of attempts and their combinations. These models are later used as a base metric for our resource adaptation model and/or learning situation. The adaptation model is also based on the three traces mentioned above and on our evaluation models. Our adaptation model automatically generates adapted paths using a state-transition model. The states represent learning situations that consume resources and the transitions between situations express the necessary conditions to pass from one situation to another. These concepts are implemented in a domain ontology and an algorithm that we have developed. The algorithm ensures two types of adaptation : (i) Adaptation of the situation and (ii) Adaptation of resources within a situation.

In order to collect traces of training for the implementation of our approaches of learner evaluation and adaptation of resources and learning situations, we conducted experiments on two groups of students in Computer Science (L2). One group in classical training and the other group in adapted training. Based on the obtained traces from the students' training sessions, we assessed learners based on our evaluation models. The results are then used to implement the adaptation in a domain ontology. The latter is implemented within oracle 11g which allows a rule-based semantic reasoning.

After comparing the results of the adapted training with those obtained from the classical one, we found an improvement in the results in terms of general average and standard deviation of the learner averages.

**Keywords:** e-learning, intelligent tutoring system (ITS), assessment, adaptation, traces.

# Table des matières

<b>Remerciements</b>	<b>2</b>
<b>Liste des tableaux</b>	<b>13</b>
<b>Liste des Figures</b>	<b>15</b>
<b>Liste des Algorithmes</b>	<b>17</b>
<b>Partie I Introduction et Positionnement</b>	<b>18</b>
1 Contexte général et motivations . . . . .	19
2 Cadre spécifique des travaux de la thèse . . . . .	21
3 Éléments fondamentaux liées au cadre spécifique . . . . .	23
3.1 Système interactif adaptatif . . . . .	23
3.2 Évaluation bienveillante et positive . . . . .	23
3.3 Notion d'Adaptation . . . . .	24
4 Objectifs et problématiques . . . . .	27
5 Plan de la thèse . . . . .	28
<b>Partie II E-Learning et Systèmes Tutoriels Intelligents</b>	<b>34</b>
<b>Chapitre 1 Systèmes Tutoriels Intelligents</b>	<b>38</b>
1.1 Introduction . . . . .	38
1.2 Définitions et caractéristiques des systèmes tutoriels intelligents . . . . .	39
1.3 Types de Système Tutoriel Intelligent . . . . .	41
1.3.1 STI basés sur les attentes des méconnaissances . . . . .	42
1.3.2 STI basés sur le modèle de traçage . . . . .	42
1.3.3 STI basés sur la modélisation par les contraintes . . . . .	43
1.3.4 STI basés sur la modélisation en réseau bayésien . . . . .	43

1.4	Évaluation des systèmes d'apprentissage en ligne en terme de performance pédagogique . . . . .	44
1.5	Évaluation des systèmes d'apprentissage en ligne en terme de fonctionnalités	48
1.5.1	Approche symbolique pour l'évaluation des plateformes d'e-learning	48
1.5.2	Utilisation de nos opérateurs de comparaison pour classer les systèmes d'e-Learning . . . . .	50
1.5.3	Exemple illustratif . . . . .	51
1.6	Exemples de quelques systèmes tutoriels intelligents . . . . .	56
1.7	L'exercice dans les plateformes de formation classique et dans les STI . . .	59
1.8	Discussion et conclusion . . . . .	62
<b>Chapitre 2 Évaluation de l'apprenant pour l'adaptation pédagogique</b>		<b>64</b>
2.1	Introduction . . . . .	64
2.2	Méthodes d'évaluation automatisées des apprenants . . . . .	65
2.3	Évaluation dans les Systèmes Tutoriels Intelligents . . . . .	67
2.4	Évaluations adaptées . . . . .	68
2.5	Systèmes d'adaptation dans les plateformes d'apprentissage en ligne . . . .	70
2.6	Adaptation dans les Systèmes Tutoriels Intelligents . . . . .	74
2.7	Bilan . . . . .	76
2.8	Conclusion . . . . .	78
<b>Partie III Évaluation et Adaptation de la stratégie d'apprentissage pour l'apprenant à base de traces</b>		<b>80</b>
<b>Chapitre 3 Modèles d'évaluation à base de traces</b>		<b>84</b>
3.1	Introduction . . . . .	84
3.2	Concepts et définitions . . . . .	85
3.2.1	Définition des Traces . . . . .	85
3.2.2	Définition de l'évaluation . . . . .	87
3.3	Modèle d'évaluation à base de traces . . . . .	88
3.3.1	Modèle d'évaluation fondé sur la durée d'interaction . . . . .	88
3.3.2	Modèle d'évaluation fondé sur le nombre de tentatives . . . . .	90
3.3.3	Modèle d'évaluation combiné fondé sur la durée et le nombre de tentatives . . . . .	92
3.3.4	Processus d'évaluation automatisée à base de traces . . . . .	94
3.4	Discussion et conclusion . . . . .	95

---

**Chapitre 4 Modèle d'adaptation de ressources et de stratégies d'apprentissage à base de traces** **96**

4.1	Introduction . . . . .	96
4.2	Processus d'adaptation à base de traces . . . . .	97
4.3	Ontologie d'adaptation du domaine de formation à base de traces . . . . .	98
4.4	Adaptation de la stratégie d'apprentissage à base de traces . . . . .	101
4.5	Discussion et conclusion . . . . .	105

**Partie IV Expérimentation et Évaluation** **107**
**Chapitre 5 Mise en œuvre de l'adaptation à base de traces par un modèle ontologique** **110**

5.1	Introduction . . . . .	110
5.2	Approche de mise en œuvre de l'ontologie . . . . .	110
5.2.1	Ontologies dans les systèmes d'apprentissage . . . . .	110
5.2.2	Gérer les données ontologiques . . . . .	114
5.2.3	Implantation de l'ontologie d'adaptation du domaine de formation . . . . .	114
5.2.4	Processus de validation de l'ontologie d'adaptation du domaine de formation . . . . .	114
5.3	Règles d'inférences . . . . .	115
5.3.1	Base de règles d'inférence d'Oracle . . . . .	116
5.3.2	Règles d'inférence métiers . . . . .	117
5.3.3	Les éléments de l'ontologie . . . . .	118
5.3.4	Règle d'inférence pour le processus d'adaptation . . . . .	121
5.3.5	Interrogation de la base de données ontologique . . . . .	123
5.4	Conclusion . . . . .	126

**Chapitre 6 Évaluation et validation** **128**

6.1	Introduction . . . . .	128
6.2	Environnement logiciel de tests . . . . .	129
6.3	Expérimentation (1) : Apprentissage avec une scénarisation classique . . . . .	129
6.4	Expérimentation (2) : Apprentissage avec une scénarisation adaptée . . . . .	134
6.4.1	Apprentissage avec une scénarisation adaptée à base de la trace de tentative . . . . .	135
6.4.2	Apprentissage avec une scénarisation adaptée à base de la trace temporelle . . . . .	138
6.4.3	Apprentissage avec une scénarisation adaptée à base de la combinaison de la trace temporelle et de tentative . . . . .	140

6.4.4	Résultats obtenus par les situations TD, TP et Test . . . . .	144
6.5	Bilan . . . . .	145
6.6	Conclusion . . . . .	146

## **Partie V Conclusion 148**

### **Chapitre 7 Conclusion Générale 150**

7.1	Bilan des contributions et apports de la thèse . . . . .	150
7.1.1	Apports bibliographiques . . . . .	151
7.1.2	Apports méthodologiques . . . . .	152
7.1.3	Apports sur l'adaptation . . . . .	154
7.2	Apports expérimentaux . . . . .	154
7.3	Apports à l'éducation numérique . . . . .	154
7.4	Limites . . . . .	155
7.4.1	Limites autour de nos propositions . . . . .	155
7.5	Perspectives . . . . .	155
7.5.1	Situation d'apprentissage composée . . . . .	156
7.5.2	Apprentissage collaboratif . . . . .	156
7.5.3	Pré-construction du profil apprenant . . . . .	156
7.5.4	Distance sémantique . . . . .	156
7.5.5	Explication et assistance . . . . .	157
7.6	Publications . . . . .	157

### **Bibliographies 159**

# Liste des tableaux

1.1	La table des opérateurs <i>prefMax</i> et <i>prefMin</i> . . . . .	50
1.2	Évaluation des fonctionnalités des plateformes d'e-Learning. . . . .	53
1.3	Résultats de <i>prefMin</i> et <i>prefMax</i> pour la catégorie « Outils de communication ». . . . .	54
1.4	Résultats du calcul de <i>prefMaxMin</i> sur l'ensemble des plateformes d'e-learning considérées. . . . .	55
1.5	Résultats du calcul de <i>prefMinMax</i> sur l'ensemble des plateformes d'e-learning considérées. . . . .	56
1.6	Description des différents STI . . . . .	58
1.7	Comparaison entre la situation de « Test » dans les plateformes de formation classique et dans les STI . . . . .	60
2.1	Tableau synthétique sur les méthodes d'adaptation existantes . . . . .	72
2.2	Tableau synthétique sur quelques méthodes d'adaptation dans les STI . . . . .	75
3.3.1	Exemple d'instanciation du modèle basé sur la trace temporelle. . . . .	90
3.3.2	Exemple d'instanciation du modèle basé sur la trace de tentative. . . . .	92
3.3.3	Exemple d'instanciation du modèle combiné de la trace temporelle et du nombre de tentatives. . . . .	93
4.4.1	Scénario de situations avec leurs ressources . . . . .	104
4.4.2	Les traces d'apprenant obtenues lors de l'exécution des situations . . . . .	105
5.1	Modèle d'évaluation et système d'adaptation à base de traces . . . . .	119
5.2	Hierarchie de classes de l'ontologie du domaine de la formation . . . . .	120
5.3	Les objets de propriétés de l'ontologie du domaine de la formation . . . . .	121
6.1	Scénario de situations avec leurs ressources . . . . .	130

---

6.2	Scores obtenus pour la situation de cours dans la formation traditionnelle. . . . .	132
6.3	Statistiques des résultats obtenus dans la formation traditionnelle. . . . .	132
6.4	Statistiques des valeurs obtenues dans l'apprentissage classique . . . . .	132
6.5	Les résultats obtenus en apprentissage classique . . . . .	133
6.6	Résultats de Cours à base de la trace de tentative . . . . .	136
6.7	Statistiques des résultats de cours obtenus sans et avec adaptation à base de la trace de tentatives . . . . .	137
6.8	Résultats de test de Cours à base de la trace temporelle . . . . .	139
6.9	Statistiques des résultats de cours obtenus sans et avec adaptation à base de la trace temporelle . . . . .	140
6.10	Résultats de Cours à base de la combinaison de la trace temporelle et de tentative	141
6.11	Statistiques des valeurs de cours obtenues sans et avec adaptation à base de la trace de temporelle et de tentatives . . . . .	144
6.12	Statistiques des valeurs de TD, TP, Test et moyenne obtenues sans et avec adaptation . . . . .	145



# Liste des Figures

1	La transition du paradigme d’enseignement du traditionnel vers le moderne [Dašić <i>et al.</i> , 2016]. . . . .	20
2	Les trois composantes fondamentales d’un système de formation adaptatif. . . . .	25
3	Modèle d’adaptation « diagnostic-prescription » . . . . .	26
4	Modèle d’adaptation « capture-analyse-sélection-présentation ». . . . .	26
5	Logique de conception, d’observation et de pilotage de l’activité. . . . .	29
6	Processus d’Évaluation à Base de Traces. . . . .	30
7	Architecture du processus d’adaptation à base de trace. . . . .	30
8	Plan du manuscrit . . . . .	31
3.1	Aperçu du système à base de traces. . . . .	86
3.2	Aperçu du déroulement du processus d’évaluation. . . . .	95
4.1	Architecture du processus d’adaptation à base de traces. . . . .	98
4.2	Fragment de l’ontologie d’adaptation du domaine de formation. . . . .	100
4.3	Diagramme État-Transition pour l’adaptation des situations et des ressources. . . . .	103
4.4	Résultat de l’adaptation des situations et des ressources à base de traces. . . . .	105
5.1	Architecture fonctionnelle du mécanisme de rétroaction. . . . .	112
5.2	Niveaux de l’ontologie d’objet d’apprentissage et de domaine [Passier et Jeuring, 2004]. . . . .	113
5.3	Inférence à base de règles dans un modèle ontologique [Chuck <i>et al.</i> , 2013] . . . . .	116
6.1	Environnement logiciel de tests. . . . .	129
6.2	Résultats des apprenants en formation traditionnelle. . . . .	131
6.3	Résultats de cours à base de la trace de tentative. . . . .	137

---

6.4	Comparaison des résultats cours des deux groupes avec et sans adaptation. . . .	137
6.5	Résultats de cours à base de la trace temporelle. . . . .	139
6.6	Comparaison des résultats cours des deux groupes avec et sans adaptation. . . .	140
6.7	Résultats de cours à base de la combinaison de la trace temporelle et de tentative.	141
6.8	Résultats du cours après adaptation à base de la combinaison de la trace temporelle et de tentative . . . . .	143
6.9	Comparaison des résultats cours des deux groupes avec et sans adaptation. . . .	144
6.10	Comparaison des résultats TD, TP, Test et moyenne générale du groupe avec et sans adaptation. . . . .	146

# Liste des Algorithmes

1	Processus d'adaptation à base de traces . . . . .	102
---	---	-----

# **Partie I**

## **Introduction et Positionnement**

## Introduction générale

Ce chapitre est une introduction générale aux différents travaux qui ont été effectués durant cette thèse. Il aligne le contexte général et les motivations derrière ces travaux de recherche de thèse pour la proposition de modèles d'évaluation de l'apprenant afin d'adapter la ressource ou la situation d'apprentissage en conséquent.

Cette dernière décennie a été marquée par l'utilisation croissante des plateformes d'apprentissage en ligne pour compléter la formation en présentiel. Cette croissance est due aux fonctionnalités offertes par ces systèmes d'apprentissage qui facilitent l'enseignement et l'apprentissage à distance en prenant en considération les besoins des apprenants et des formateurs. De plus en plus, l'enseignement utilisant les plateformes en ligne se concentre sur les résultats plutôt que sur les structures. Le succès des étudiants est devenu l'un des principaux facteurs de qualité de l'enseignement, et des programmes en ligne. Bien que le succès des apprenants ait été défini de diverses façons, la plupart des définitions incluent l'idée de persévérance jusqu'à l'achèvement du programme par l'apprenant. Ainsi, une rétention accrue devient l'objectif de nombreux efforts d'évaluation et d'amélioration de la qualité d'une formation.

Pour diminuer les abandons des apprenants et assurer leur persévérance, il est nécessaire d'avoir de nouveaux modèles d'évaluation précoces, constructifs et adaptés qui prennent en considération des indicateurs de performance autres que la réponse de l'apprenant. Cette évaluation est dite adaptée car elle permet une régulation de la formation en adaptant la ressource et/ou la situation pédagogique à l'apprenant en cas où il échoue dans celle-ci.

## 1 Contexte général et motivations

Afin de soutenir le processus d'apprentissage dans l'enseignement traditionnel ou en remplacement d'un tel enseignement, les nouvelles technologies du numérique sont devenues une partie essentielle du système. Dans cette direction, le multimédia, les réseaux et l'ingénierie de logiciels ont conduit à l'émergence d'une nouvelle génération de systèmes informatiques pour l'apprentissage en ligne dite Technologie d'Apprentissage Avancée (Advanced Learning Technology-ALT) [Sandholtz *et al.*, 1999; Murray, 1999; Phobun et Vicheanpanya, 2010; Rosatelli et Self, 2004; Zhang, 2005].

Le nouveau paradigme pédagogique est orienté apprenant ou, en d'autres termes, axé sur l'apprenant [Bruce et Bruce, 1993; Handa *et al.*, 2009; Reigeluth *et al.*, 2015]. L'apprenant est placé au centre du système d'apprentissage, tandis qu'aux alentours sont positionnées les ressources d'apprentissage (personnes, connaissances, technologies, médias, organisations, etc.) en termes de temps, de lieu et de mode d'apprentissage comme l'illustre la Figure 1.

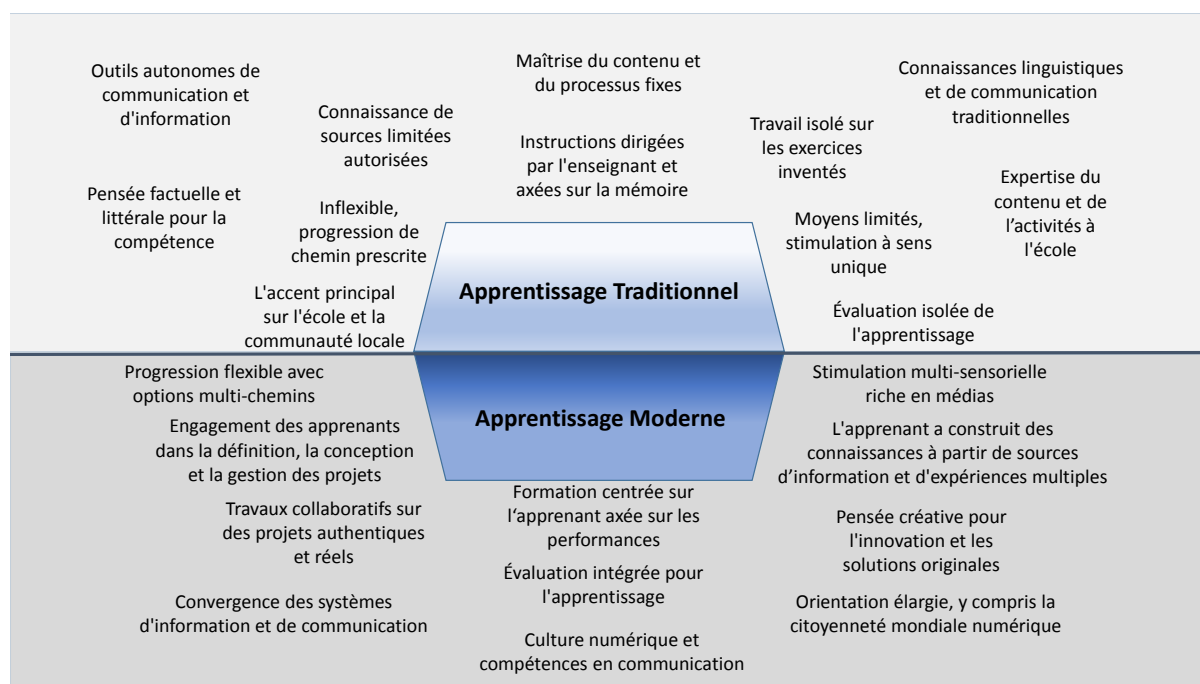


FIGURE 1 – La transition du paradigme d'enseignement du traditionnel vers le moderne [Dašić *et al.*, 2016].

Parmi les nombreux systèmes d'apprentissage centrés apprenant, on peut considérer les systèmes e-learning qui sont des applications directes des nouvelles technologies du numérique qui utilisent des fonctionnalités de base pour l'administration des apprenants et la gestion des contenus pédagogiques [Markowska Kaczmar *et al.*, 2010]. Cependant, ces systèmes ne peuvent générer que du contenu pédagogique statique où l'apprenant reçoit passivement l'information transmise par le formateur. Pour cela, l'intégration des fonctionnalités interactives, adaptables et intelligentes était nécessaire pour la génération du contenu d'apprentissage d'une manière dynamique qui engage l'apprenant dans un processus d'apprentissage actif pour mieux comprendre, analyser et intégrer des contenus. Cet aspect de génération de contenus dynamique a été adopté et intégré dans le développement des noyaux des Systèmes Tutoriels Intelligents (STI) [Torres et Sentí, 2017].

Par ailleurs, il existe différentes catégories d'activités ouvertes, flexibles et distribuées dans le processus d'enseignement et d'apprentissage, notamment : Hypermedia Adaptative (Adaptive Hypermedia-AH), Environnements d'Apprentissage Intelligents (Intelligent Learning Environments-ILE), Système Tutoriel Intelligent-STI (Intelligent Tutoring system-ITS), etc. [Brusilovsky, 2000; Brusilovsky *et al.*, 1998; Brusilovsky et Peylo, 2003; Dašić *et al.*, 2010, 2007; Phobun et Vicheanpanya, 2010; Reigeluth *et al.*, 2015; Wasson, 1997]. Les Systèmes Tutoriels Intelligents représentent notre champ d'étude.

Les environnements d'apprentissage basés sur le web, permettent l'enseignement assisté via l'accès à l'information et ses caractéristiques organisationnelles des ressources dans le domaine du multimédia. Dans ces systèmes, les auteurs évitent la création de contenus d'apprentissage

statiques présentés à l'apprenant de manière linéaire, en raison de l'interdépendance et de liens conditionnels entre les différentes pages web [Brent *et al.*, 2017]. Souvent, les auteurs créent plusieurs versions de ressources d'apprentissage afin que le système puisse proposer à l'apprenant la ressource la plus appropriée à son profil. Cela conduit au concept d'apprentissage connu sous le nom **d'adaptation de contenus ou de ressources**. Cette adaptation doit tenir compte des différents éléments de mesures de l'apprenant tels que les indicateurs de performance personnalisés obtenus des données collectées pendant les sessions d'apprentissage, les styles d'apprentissage, les objectifs pédagogiques et le niveau de connaissances de l'apprenant [Djouad *et al.*, 2010; Mislevy *et al.*, 2012; Sterbini et Temperini, 2009; Felder et Silverman, 1988; Felder et Spurlin, 2005; Bloom's, 1965; Gorgone *et al.*, 2002]. Peu d'approches utilisent les scores des apprenants calculés pendant ou après une situation d'apprentissage pour mesurer leur performance pour permettre une adaptation des contenus ou des ressources pédagogique en conséquent [Graf, 2007; Rey-López *et al.*, 2009].

Les méthodes d'évaluation existantes sont statiques et reposent sur des modèles linéaires. Dans ces méthodes, la notation de l'apprenant ne prend en considération que sa réponse et néglige d'autres métriques du modèle de l'apprenant et/ou de son modèle de connaissances qui peuvent révéler d'autres indicateurs de performance. Par exemple, le temps de résolution d'un problème, le nombre de tentatives, la qualité de la réponse, etc. Ces éléments servent à détecter les traits du profil, le comportement ainsi que les troubles d'apprentissage de l'apprenant. Ces paramètres additionnels sont vues dans nos travaux de recherche comme des traces d'apprentissage produites par l'apprenant durant une situation ou un contexte pédagogique donné. Le processus d'évaluation fondé sur les traces permet de calculer des indicateurs pertinents qui peuvent être utilisés dans la personnalisation de l'analyse [Settouti *et al.*, 2009c], étant donné que les traces considérées sont produites par l'apprenant.

Le contexte général de nos travaux s'inscrit dans le domaine d'adaptation de la ressource et/ou de la situation d'apprentissage en se basant sur l'évaluation de l'apprenant.

## 2 Cadre spécifique des travaux de la thèse

Dans cette thèse, nous évaluons le travail de l'apprenant dans les plateformes d'apprentissage en ligne et en particulier dans les systèmes tutoriels intelligents (STI). Dans ces systèmes, l'apprenant interagit avec des ressources dans une situation d'apprentissage. L'historique d'interaction de l'apprenant avec les composants du système au cours d'une activité pédagogique fait partie des paramètres utilisés pour l'évaluation du travail de l'apprenant.

Le cadre de l'activité faite par l'apprenant rentre dans un contexte de scénarisation de situations. Cette dernière est paramétrée par un expert du domaine ou un formateur. L'évaluation de l'apprenant implique la notion de données produites en se basant sur les métriques fixées par

l'expert ou le formateur et les traces produites par l'apprenant durant son activité, par exemple la durée et le nombre d'essais effectués pour résoudre un problème donné.

Une évaluation de l'apprenant peut réguler un scénario défini ou prévu par un formateur en proposant ou en adaptant une meilleure situation et/ou une ressource qui répondent aux besoins de l'apprenant selon son évaluation obtenue. Cela introduit la notion de scénario émergent qui est le résultat d'une adaptation de situations et/ou de ressources consommées dans une situation. Un système qui adopte ces notions est dit « système interactif adaptatif ». Nous définissons ultérieurement les notions d'« système interactif adaptatif », « évaluation » et « adaptation ».

Pour la mise en place d'un apprentissage adaptatif, nous avons besoin de connaître les caractéristiques de l'apprenant dans le système d'apprentissage en ligne (modèle de l'apprenant), les situations pédagogiques disponibles et leurs ressources associées (modèle pédagogique). Ces entrées sont aussi nécessaires aux modèles d'évaluation et au processus d'adaptation de la situation et/ou de la ressource pédagogique. Les paramètres sont résumés comme suit :

- dans le modèle d'apprenant, nous avons besoin de savoir son profil (son niveau, ses compétences, ses objectifs, etc.), son historique (la durée et le nombre d'essais effectués durant une situation d'apprentissage) ;
- dans le modèle pédagogique, nous avons besoin de connaître le type de présentation de la stratégie d'apprentissage : (i) socratique qui consiste en un questionnement qui favorise chez l'apprenant la naissance des idées dont il serait porteur. Cette stratégie vise divers types de connaissances et favorise l'apprentissage des connaissances conceptuelles, procédurales, factuelles ou métacognitives en rendant l'apprenant actif [Garlikov, 2001], ou (ii) simulation pour savoir comment et quand présenter les contenus ou ressources dans le système ;
- en ce qui concerne le modèle expert, nous avons besoin de connaître les paramètres de configuration exigés par l'expert ou le formateur, comme les objectifs fixés dans une activité, la note optimale proposée pour atteindre un objectif, le temps prévu et le nombre de tentatives autorisés pour résoudre un problème donné.

L'évaluation nécessite comme entrées les paramètres susmentionnés pour savoir si l'apprenant a réussi ou a échoué une situation d'apprentissage. L'adaptation se base sur l'évaluation de l'apprenant et les métriques fixées par l'expert ou le formateur pour la définition des règles d'adaptation de la ressource ou de la situation pédagogique.

Les trois modèles (modèle de l'apprenant, modèle pédagogique et modèle de l'expert) mentionnés ci-dessus, sont nécessaires à la construction d'un modèle d'évaluation et d'un système d'adaptation de la ressource et/ou de la situation d'apprentissage ; ils représentent aussi la base de nos contributions.



## 3 Éléments fondamentaux liés au cadre spécifique

### 3.1 Système interactif adaptatif

Les plateformes d'apprentissage en ligne font partie de la famille des systèmes interactifs qui permettent les interactions entre ses différentes composantes humaines et logiciels. La notion d'interactivité dans ces systèmes est vue comme une construction liée au processus de communication en créant un médiateur qui permet l'échange de messages entre les composantes. La communication est faite d'une manière synchrone ou asynchrone [Kioussis, 2002]. Effectivement, durant l'exécution de ces systèmes interactifs, différentes trajectoires peuvent être empruntées par les utilisateurs afin d'arriver à leurs objectifs. La trajectoire empruntée n'est pas toujours optimale et dépend fortement du contexte de l'utilisateur. Pour cela, une adaptation est nécessaire au cours d'une exécution du système interactif. L'approche proposée facilite le pilotage d'un scénario dans le système en permettant l'adaptation durant son exécution. Elle prend en considération les interactions entre acteurs et le système et assure la cohérence dans ce dernier, et ce, en adoptant le concept de situation [Pham *et al.*, 2013].

La notion de scénarisation dans les systèmes interactifs est basée sur des structures scénarisées. Ces dernières ne prennent pas en considération l'aspect complexe, dynamique et adaptable des scénarii. En effet, les interactions et les actions dans ces structures sont imprédictibles. Pour cela, dans notre contexte spécifique, nous nous intéressons à une nouvelle structure, appelée « *situation* » [Pham, 2013]. Cette structure prend en compte les acteurs, les ressources, les interactions, la gestion de la cohérence et facilite l'adaptation. De ce fait, nous nous basons sur une scénarisation ou scénarii à base de situations contextualisées.

Dans nos travaux de recherche, nous proposons donc d'évaluer cette situation dans un contexte d'apprentissage en ligne, en évaluant les apprenants et les ressources impliquées dans l'exécution de cette situation tout en respectant les pré-conditions et les post-conditions nécessaires afin d'atteindre les objectifs locaux et globaux voulus. En conséquence, l'évaluation de l'apprenant nous permet d'adapter la ressource et la stratégie d'apprentissage à ce dernier.

### 3.2 Évaluation bienveillante et positive

Concernant l'évaluation bienveillante et positive, la loi d'orientation précise qu'il faut : « *privilégier une évaluation positive, simple et lisible, valorisant les progrès, encourageant les initiatives et compréhensible par les familles. En tout état de cause, doit permettre de mesurer le degré d'acquisition des connaissances et compétences ainsi que la progression de l'élève* »<sup>1</sup>.

<sup>1</sup> [www.education.gouv.fr/cid102387/loi-n-2013-595-du-8-juillet-2013-d-orientation-et-de-programmation-pour-la-refondation-de-l-ecole-de-la-republique.html](http://www.education.gouv.fr/cid102387/loi-n-2013-595-du-8-juillet-2013-d-orientation-et-de-programmation-pour-la-refondation-de-l-ecole-de-la-republique.html)

Cette loi a pour objectif de fonder une évaluation pédagogique basée sur une bienveillance rigoureuse et une notation positive. Cependant, la loi promulguée en 2014 vise à favoriser la réussite des apprenants et « *améliorer l'efficacité des apprentissages et la confiance des élèves* »<sup>2</sup>. L'évaluation est redéfinie comme l'appréciation de l'acquisition des compétences et des connaissances de la part de l'apprenant à travers un « bilan détaillé de son travail », capable en même temps de « lui donner les moyens de progresser et de résoudre ses difficultés scolaires ». Ce genre d'évaluation permet d'instaurer un climat de confiance et d'éloigner l'apprenant du stress et de la peur d'une épreuve, tout en permettant de « *prélever des informations en retour éclairantes, en ce sens qu'elles vont permettre d'apprécier une situation existante par référence à une situation but* » [Hadji, 2015].

L'évaluation du travail en classe est faite par une évaluation formative [Nicol et Macfarlane-Dick, 2006], dont la dimension constructive a pour ambition d'apporter des réponses à l'enseignant s'interrogeant sur les maîtrises, les difficultés et les progrès d'apprentissage réels des apprenants, tout en lui offrant des indicateurs pour adapter son action et ses programmes futurs. Ladite évaluation permet aux apprenants une régulation et un guidage de l'activité d'apprentissage basée sur la rétroaction de l'enseignant.

Par ailleurs, l'utilisation des technologies du numérique [Laurillard, 2013] dans le domaine d'enseignement a participé au développement des systèmes d'apprentissage en ligne, système tutoriel intelligent, les Moocs, etc. Ces systèmes permettent aux apprenants une formation à distance adaptée à leurs besoins. À cet effet, des systèmes d'évaluation en ligne ont été développés permettant une notation pendant une activité d'apprentissage [Wilhelm, 2009; Scheuermann et Björnsson, 2009; Kozma, 2009; Gikandi *et al.*, 2011]. Cependant, la majorité des systèmes d'évaluation développés sont statiques et ne prennent pas en considération le statut dynamique de l'apprenant et de ses caractéristiques changeantes durant une activité d'apprentissage ou de test [Nicol, 2007; Crisp, 2009; Guo, 2013a].

L'objectif de nos travaux de thèse est de fournir une évaluation dynamique afin d'adapter la ressource pédagogique et/ou la stratégie d'apprentissage. Dans ce contexte, peu de travaux ont été développés [Graf, 2007; Rey-López *et al.*, 2009; Burgos *et al.*, 2007].

### 3.3 Notion d'Adaptation

L'adaptativité dans les systèmes d'apprentissage peut être définie de manière générale comme la capacité d'un système à ajuster les instructions en fonction des aptitudes et/ou des préférences des apprenants. L'objectif est d'agir sur les caractéristiques identifiées de l'apprenant et d'améliorer l'efficacité et l'efficience de l'apprentissage [Oxman *et al.*, 2014; Vandewaetere *et al.*, 2011]. Outre l'exigence à la diversité de l'apprenant, les systèmes adaptatifs

---

<sup>2</sup>Circulaire 2014-068 du 20-05-2014

offrent également une interactivité car ils ont pour objectif de modéliser les formateurs humains et leurs compétences pédagogiques et expertes [Oxman *et al.*, 2014].

La base technologique de l'enseignement adaptatif peut être identifiée par trois éléments partagés par tous les systèmes adaptatifs [Brusilovsky, 2012; Knutov *et al.*, 2009; Vandewaetere *et al.*, 2011] comme illustré sur la Figure 2 : (1) le modèle de l'apprenant représente la source de l'adaptation, (2) le modèle de domaine décrit la cible d'adaptation et (3) la connexion établie entre le modèle de l'apprenant et le modèle de domaine, mise en œuvre par le modèle d'adaptation. L'introduction d'autres composants dans l'architecture du système est également possible [Blake, 2009; Šimko *et al.*, 2010]. Néanmoins, les trois composants susmentionnés sont une condition préalable nécessaire pour que chaque système adaptatif puisse identifier les caractéristiques individuelles de l'apprenant et décider quelle, quand et comment une instruction adaptée sera livrée à un apprenant particulier.

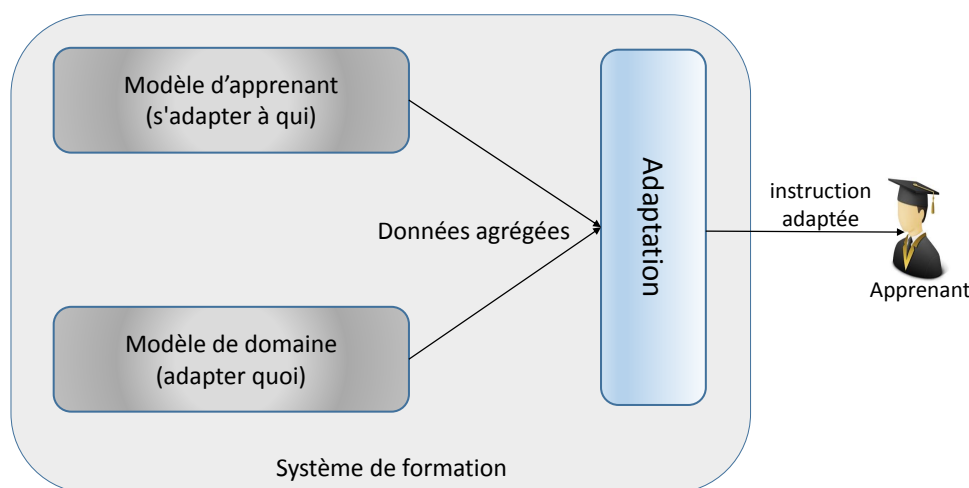


FIGURE 2 – Les trois composantes fondamentales d'un système de formation adaptatif.

Le mécanisme d'adaptation d'un système d'enseignement est chargé d'identifier le niveau actuel de connaissances, de besoins, d'objectifs ou d'intérêts d'un apprenant [Knutov *et al.*, 2009] et de fournir un soutien éducatif approprié en retour. Cela se traduit traditionnellement par le simple modèle d'adaptation « diagnostic-prescription » [Shute et Zapata-Rivera, 2012]. Le développement de l'enseignement adaptatif est basé sur deux approches, à savoir « l'enseignement macro-adaptatif » et l'« aptitude-traitement d'interactions » [Park et Lee, 2003]. Ce modèle peut être considéré comme celui dans lequel la détermination des traits de l'apprenant se produit une seule fois lors de l'apprentissage, notamment dans la phase initiale, par exemple en utilisant une procédure de pré-test ou en classant les apprenants dans des clusters stéréotypés et en décidant du scénario d'instructions approprié [Rich, 1979] comme le montre la Figure 3. Le système fournit donc une adaptation basée sur un ensemble inchangé de caractéristiques de l'apprenant tout au long de l'enseignement, mais ne reflète pas le contexte pratique de l'apprentissage. Le modèle traditionnel a été étendu à un modèle plus approprié,

à quatre étapes cycliques « capture-analyse-sélection-présentation » [Shute et Zapata-Rivera, 2012]. Comme résumé dans la Figure 4, le système surveille de près l'apprenant pendant son interaction, rassemble les données et les enregistre. Les données rassemblées sont analysées et interprétées pour inférer les caractéristiques de l'apprenant et mettre à jour celles existantes si nécessaire. Les données pertinentes de l'apprenant sont choisies et utilisées pour initier un processus adaptatif; les instructions appropriées sont alors transmises à l'apprenant. Comme on peut le remarquer, ce modèle tient compte du fait que les caractéristiques des apprenants changent pendant les sessions de formation.

La notion de suivi des apprenants pendant l'utilisation du système et l'exécution de mises à jour en temps opportun des caractéristiques identifiées font partie des deux autres approches, à savoir : « l'enseignement macro-adaptatif » et les « aptitude-traitement d'interactions », suggérées dans [Park et Lee, 2003]. Ces approches sont appelées « instruction micro-adaptative », et sont implémentées dans les systèmes de tutorat intelligents, de systèmes hypermédia adaptative, qui utilisent des éléments hypertextes/multimédias pour adapter l'enseignement.

Nous nous basons sur l'approche « instruction micro-adaptative » pour construire notre modèle d'adaptation en utilisant les traces d'apprenant capturées pendant une activité d'apprentissage.

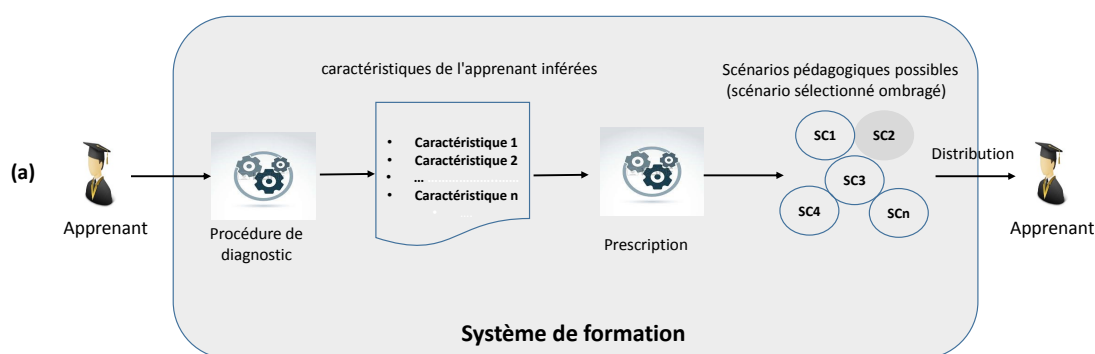


FIGURE 3 – Modèle d'adaptation « diagnostic-prescription »

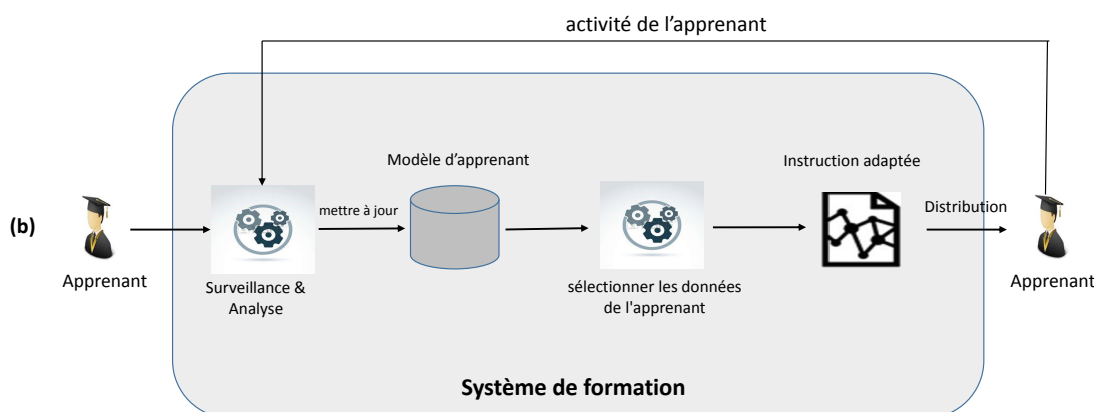


FIGURE 4 – Modèle d'adaptation « capture-analyse-sélection-présentation ».

## 4 Objectifs et problématiques

Dans le contexte général décrit ci-dessus, nous considérons les plateformes d'apprentissage en ligne. Dans la réalité, les utilisateurs apprennent de façons différentes et à des rythmes différents. Les systèmes actuels offrent des approches uniformes qui ne tiennent pas compte de cette diversité d'apprentissage. Les apprenants peuvent plus au moins adhérer au processus d'apprentissage pédagogique. Cela explique certains **échecs**, certains **abandons** ou **insatisfactions** dans les formations [Park et Choi, 2009].

La problématique que nous proposons de traiter dans nos travaux visent à introduire une dynamique de personnalisation et d'adaptation afin d'aider l'apprenant durant son apprentissage en lui offrant une formation adaptée à son niveau d'acquisition de connaissances. Notre approche propose de respecter autant que possible le processus d'apprentissage de l'apprenant et ses caractéristiques pour l'acquisition des connaissances. Pour cela, il faut que les mécanismes que nous proposons puissent analyser les traces d'apprentissage de l'apprenant, ses données, ses performances en acquisition durant ses activités d'apprentissage et puissent s'adapter au profil pédagogique et au modèle de l'apprenant que nous établissons dynamiquement.

L'adaptation se base sur le **profil pédagogique de l'apprenant**, le **modèle d'apprenant** et le **modèle de domaine**. Le profil sera reconstruit (calculé et mis à jour) en se basant sur les traces capturées dans le système lors des activités d'apprentissage et des interactions entre les modèles susmentionnés. Ce profil est donc le support permettant d'adapter l'apprentissage [Canut *et al.*, 2016, 2015]. Nous traçons dans le système une sorte de trajectoire d'apprentissage où chaque nœud est une ressource ou une situation d'apprentissage et notre mécanisme d'adaptation naviguera d'un nœud vers un autre ou d'une ressource vers une autre tout en vérifiant l'acquisition des connaissances en adaptant la ressource et/ou la stratégie d'apprentissage pour atteindre un objectif pédagogique [Chachoua *et al.*, 2016c].

L'adaptation pédagogique reposent donc sur un processus qui prend en considération plusieurs paramètres et caractéristiques de l'apprenant et de la ressource, à savoir le processus d'évaluation de l'apprenant dans une activité d'apprentissage. L'évaluation doit prendre en compte tous les paramètres pour inférer une stratégie d'apprentissage adaptée au profil calculé ou reconstruit de l'apprenant. Cela nous a amené à envisager des modèles ou des approches d'évaluation [Chachoua *et al.*, 2016d]. En conclusion, les travaux de thèse présentés dans ce manuscrit s'attachent à apporter des réponses aux questions suivantes :

- Qui évaluons-nous ?
- Comment évaluons-nous ?
- Quelles techniques utilisées pour faire l'évaluation ?
- De quelles données avons-nous besoin pour évaluer et comment les collecter ?

- Pourquoi évaluons-nous ?
- Est-ce que l'adaptation tient compte de la situation d'apprentissage actuelle et/ou du profil actuel d'apprenant ou d'autres paramètres d'évaluation ?
- Quel est l'impact du processus d'évaluation sur l'adaptation de l'apprentissage ?

## 5 Plan de la thèse

Nous détaillons les composants principaux nécessaires à la mise en place de l'adaptation pendant l'exécution d'une situation d'apprentissage dans un système d'apprentissage en ligne en se basant sur nos modèles d'évaluation fondés sur les traces du déroulement ou de l'exécution d'une activité. La Figure 5 décrit les composantes nécessaires à nos travaux de thèses, et ce, durant les différentes étapes (avant, pendant et après) d'exécution d'une activité d'apprentissage en ligne. Les différentes logiques d'exécution sont énumérées comme suit :

1. Logique de l'activité : est représentée par le modèle de l'apprentissage qui nous permet de faire de l'observation et du pilotage. Ce modèle d'apprentissage peut être conçu par un concepteur ou un expert du domaine ; il permet aux apprenants d'interagir avec le système. Dans nos travaux nous considérons un modèle d'apprentissage à base de situations [Pham, 2013] ;
2. Logique d'observation de l'activité : dans cette étape nous associons des observateurs aux actions, interactions et événements qui surviennent durant l'exécution de l'activité. Ces observateurs nous permettent de capturer les traces essentielles qui seront modélisées sous forme de m-Trace. La Figure 6 présente notre processus d'évaluation à base de traces, détaillé dans le chapitre 3 ;
3. Logique de pilotage de l'activité : l'étape précédente d'observation nous a permis le développement d'une couche de décision de pilotage. Le Pilotage de l'activité est une phase d'adaptation de la ressource et/ou de la stratégie de l'apprentissage. La Figure 7 montre notre processus d'adaptation à base de traces, décrit en détail dans le chapitre 4.

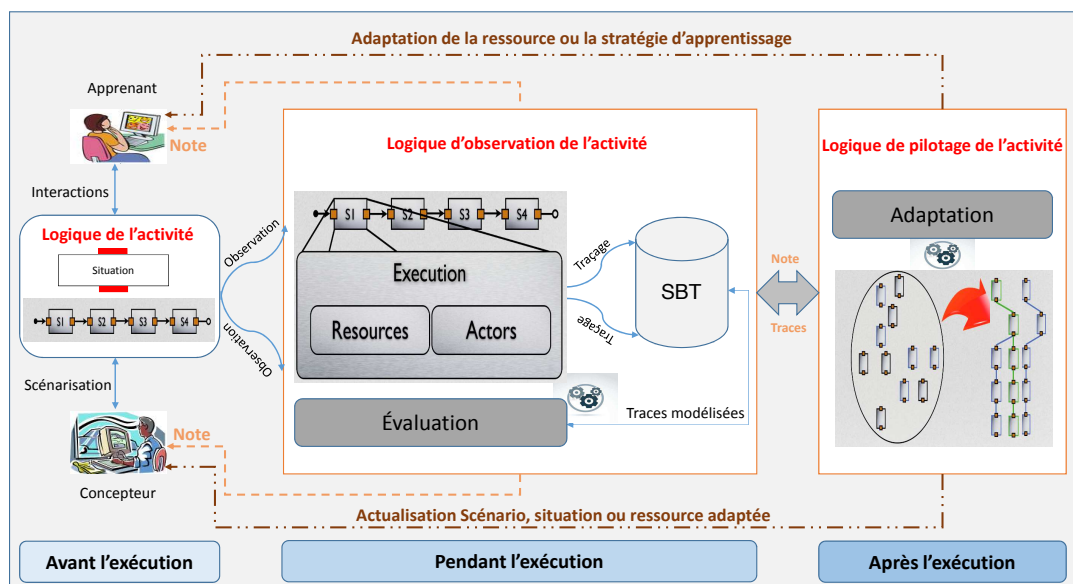


FIGURE 5 – Logique de conception, d'observation et de pilotage de l'activité.

Dans un système d'apprentissage, les apprenants exécutent des scénarii à base de situations pour apprendre. Les différentes situations sont définies par un concepteur avec une logique d'exécution que les apprenants suivent durant l'apprentissage. Cependant, cette scénarisation prévue ne répond aux besoins et aux objectifs des apprenants que dans le cas d'un profil d'apprenant statique. Dans nos travaux, nous considérons que le profil et/ou le modèle de l'apprenant est dynamique c.à.d il change avec le changement des caractéristiques ou des paramètres au cours de l'acquisition des connaissances de l'apprenant. Les paramètres pris en considération dans nos travaux sont les traces laissées dans le système après une activité pédagogique, à savoir :

1. la trace temporelle : le temps passé par l'apprenant pour résoudre un problème ;
2. la trace de tentative : le nombre de fois qu'un apprenant essaie de résoudre le problème.

Leur combinaison est également prise en compte [Chachoua *et al.*, 2016d]. Ces traces sont modélisées par la suite et utilisées dans nos modèles d'évaluation automatisés ci-dessous :

- Modèle d'évaluation fondé sur la durée d'interaction ;
- Modèle d'évaluation fondé sur le nombre de tentatives ;
- Modèle d'évaluation combiné fondé sur la durée et le nombre de tentatives.

Notre processus d'évaluation ne sert pas uniquement à préciser si l'apprenant a réussi l'activité d'apprentissage ou a acquis la compétence requise mais il sert aussi à mesurer la qualité de l'apprentissage en différenciant entre des apprenants qui ont répondu correctement à un exercice selon leurs traces.

Le processus d'évaluation automatisé et les modèles d'évaluation susmentionnés sont décrits en détail dans le chapitre 3. Enfin, le processus d'évaluation automatisée est suivie par une dernière phase qui sert à calculer des indicateurs de performance en termes d'acquisition de connaissances pour chaque apprenant afin d'adapter les situations de formation et/ou la stratégie d'apprentissage. Pour ce faire, une modélisation ontologique des concepts en termes de classes et propriétés de notre domaine est requise. L'ontologie permet d'exprimer des règles logiques sur lesquelles un raisonnement est construit pour adapter les ressources et les situations. Le processus d'adaptation à base de règles ontologiques reçoit comme entrée les traces (durée et nombre de tentatives), la note calculée par le *modèle d'évaluation* et la ressource actuelle dans la situation actuelle, le résultat produit est la prochaine ressource et/ou la situation adaptées [Chachoua *et al.*, 2016c]. Le processus d'adaptation à base de traces est détaillé dans le chapitre 4.

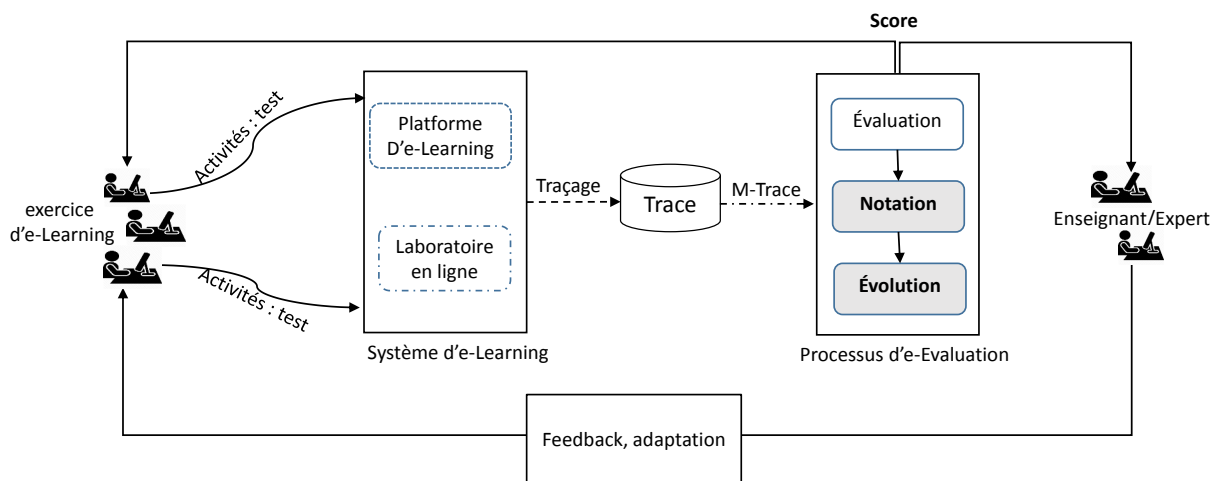


FIGURE 6 – Processus d'Évaluation à Base de Traces.

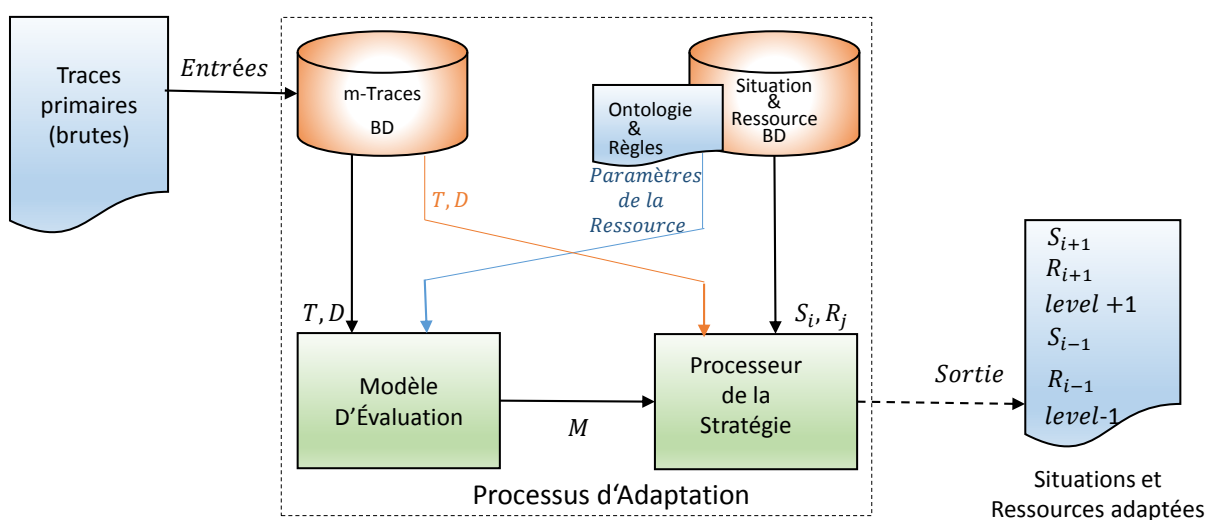


FIGURE 7 – Architecture du processus d'adaptation à base de trace.

Le manuscrit de thèse se compose de cinq parties comme illustré sur la Figure 8.



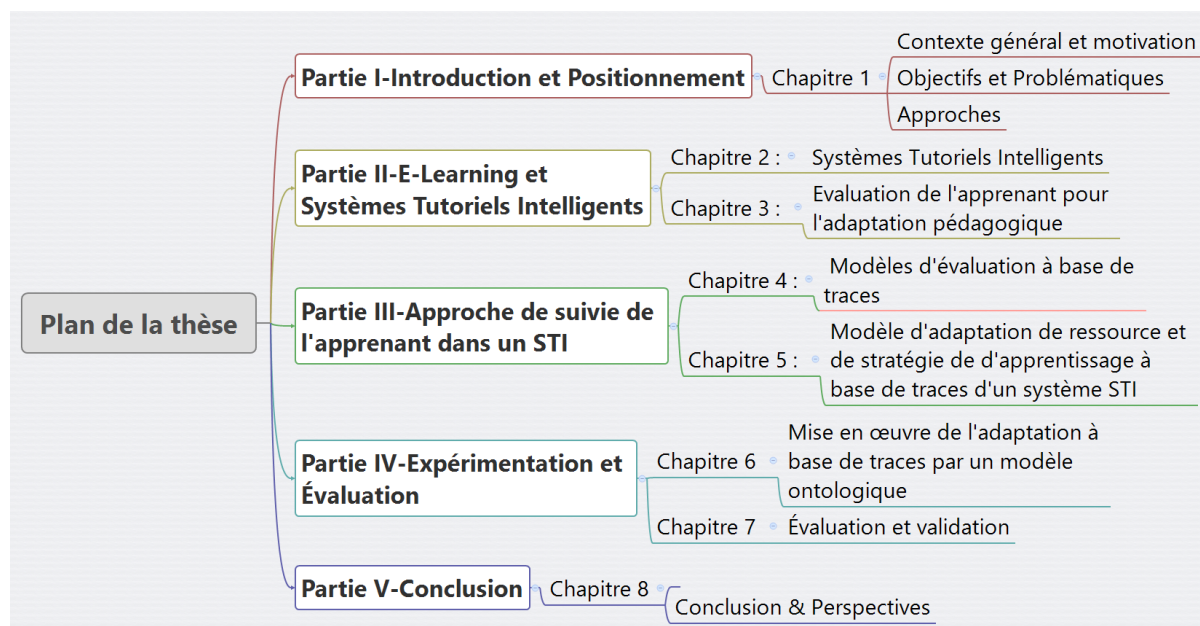


FIGURE 8 – Plan du manuscrit

La première partie du manuscrit est consacrée à la présentation du contexte générale de nos travaux, nos motivations, les objectifs et problématiques de recherche et les approches considérées dans nos contributions.

Dans la deuxième partie, le chapitre 1 et 2 sont consacrés à l'éclaircissement des éléments qui ont conduit à nos contributions. Ces chapitres présentent une analyse de l'état de l'art des problématiques scientifiques de la thèse, une analyse des systèmes et des approches existantes pour l'évaluation de l'apprenant et l'adaptation dans les systèmes d'apprentissage en ligne.

Cette analyse a permis de déterminer les éléments manquants et les verrous à lever pour satisfaire les besoins et les critères de notre cahier des charges. Toujours dans le chapitre 1 de l'état de l'art et d'après l'étude que nous avons effectuée sur les systèmes tutoriels intelligents et leur architecture basée sur le modèle de l'apprenant, le modèle pédagogique et le domaine d'expert, nous avons identifié le verrou scientifique qui portent sur la reconstruction des modèles susmentionnés dans le cas d'un profil d'apprenant dynamique qui change durant et après une situation d'apprentissage. Dans le chapitre 2 de l'état de l'art, nous avons identifié un autre élément qui porte sur l'évaluation dans les plateformes d'apprentissage en ligne. Les systèmes d'évaluation automatisés sont sous-utilisés car il est impossible d'avoir à disposition un moteur d'évaluation d'une activité d'apprentissage qui fonctionnera dans différents contextes. Dans les systèmes tutoriels intelligents, seuls quelques travaux ont abordé l'évaluation entièrement intégrée. Les méthodes d'évaluation proposées dans ces systèmes sont statiques. Le verrou scientifique consiste à tenir compte de la dynamique du profil.

Enfin, adapter la ressource et/ou la situation d'apprentissage en se basant sur les caractéristiques dynamiques du modèle de l'apprenant ou sur son évaluation reste toujours un verrou à considérer.

Nous nous intéressons aux évaluations adaptées qui nous permettent une mise en œuvre de l'adaptation pédagogique à base du calcul de la performance d'acquisition de connaissances par l'apprenant. L'analyse des systèmes d'adaptation dans les plateformes d'apprentissage en ligne et en particulier dans les systèmes tutoriels intelligents nous a permis de relever des carences que nous avons prises en considération pour définir notre modèle d'adaptation de la ressource et de la stratégie d'apprentissage.

La troisième partie est constituée de deux chapitres. Le chapitre 3 présente notre première contribution. Comme nous l'avons mentionné précédemment, l'analyse de l'état de l'art a montré que la reconstruction du modèle de l'apprenant, modèle pédagogique, domaine d'expert en se basant sur un profil dynamique selon une évaluation dynamique de l'acquisition de connaissances est un élément manquant.

Ainsi dans le même chapitre, le verrou que nous avons identifié concerne la modélisation des traces d'apprentissage pour une évaluation adaptée aux caractéristiques dynamiques de l'apprenant. Nous avons proposé d'étendre cette modélisation en intégrant une nouvelle catégorie de données appelée « données sémantiques ou inférées pour l'adaptation » qui sont déduites des règles ontologiques qui sont nécessaires à l'adaptation de l'activité de l'apprenant.

Cette nouvelle approche permet d'utiliser les données relatives aux caractéristiques d'interaction de chaque apprenant. Ensuite, nous proposons d'utiliser les traces laissées par les apprenants dans le système d'apprentissage pour reconstruire et mettre à jour le modèle de l'apprenant, le modèle pédagogique et le domaine d'expert. La contribution que nous avons présentée est un modèle d'évaluation qui permet d'utiliser les traces d'apprentissage et les données qui caractérisent l'activité de l'apprenant dans le système d'apprentissage. Nous avons proposé des modèles génériques d'évaluation à base des traces produites par les apprenants durant une session d'apprentissage. Ces modèles permettent par la suite une adaptation de la ressource et/ou de la situation d'apprentissage à chaque apprenant, et ce, grâce à la reconstruction du modèle de l'apprenant. Cette reconstruction est basée sur la modélisation des traces nécessaires et pertinentes à nos modèles d'évaluation.

Le même chapitre présente en détail nos contributions concernant l'évaluation de l'apprenant à base de traces, à savoir : (i) modèle d'évaluation fondé sur la durée d'interaction, (ii) modèle d'évaluation fondé sur le nombre de tentatives et (iii) modèle d'évaluation combiné fondé sur la durée et le nombre de tentatives.

Le chapitre 4 est consacré à nos contributions à l'adaptation de ressources et de stratégies d'apprentissage à base de trace dans les systèmes STI avec l'algorithme que nous avons proposé. Dans cette thématique, l'état de l'art a montré que l'adaptation de la stratégie d'apprentissage à base de l'évaluation de l'apprenant n'est pas bien établie dans les systèmes d'apprentissage en ligne. Nous proposons un modèle d'adaptation de la ressource et/ou de la stratégie d'apprentissage qui communique avec notre modèle d'évaluation. Ce modèle a la capacité de proposer à l'apprenant une situation ou une ressource adaptée à ses résultats d'évaluation dans

les systèmes STI. Dans ces derniers, l'adaptation a été intégrée en se basant sur nos trois modèles d'évaluation à base de traces. Pour ce faire, le domaine de formation a été décrit par une ontologie de description en termes de classes et concepts qui prend en considération les traces laissées par un apprenant après une situation d'apprentissage. Un mécanisme de raisonnement sur l'ontologie à base de règles a été aussi proposé en utilisant un modèle d'état-transition et un algorithme qui permettent de générer des trajectoires adaptées aux apprenants.

La quatrième partie contient deux chapitres. Elle concerne la preuve de concepts et l'évaluation des propositions présentées dans la partie précédente. Le chapitre 5 présente la mise en œuvre de l'adaptation à base de traces par un modèle ontologique à base de situation qui met en évidence les problèmes mentionnés dans les chapitres 1 et 2. Le chapitre 6 concerne l'expérimentation de l'adaptation de la ressource et/ou de la situation d'apprentissage à l'apprenant en se basant sur l'évaluation à base de traces liées au domaine de la formation à base de situations. Pour cela, nous avons choisi l'apprentissage du langage SQL à base de situations. L'apprenant a accès à plusieurs situations d'apprentissage (cours, travaux dirigés (TD), travaux pratiques (TP) et test) qui utilisent des ressources pédagogiques et il est évalué sur son acquisition de connaissances. En se basant sur cette évaluation, une adaptation est proposée suivant son niveau de compétence acquise.

Enfin, dans le chapitre 7, nous développons les principales conclusions aux travaux de recherche menés et nous identifions les perspectives scientifiques intéressantes à explorer dans la continuité de la thèse.

## **Partie II**

# **E-Learning et Systèmes Tutoriels Intelligents**



# Table des matières

---

<b>Chapitre 1 Systèmes Tutoriels Intelligents</b>	<b>38</b>
1.1 Introduction . . . . .	38
1.2 Définitions et caractéristiques des systèmes tutoriels intelligents . . . . .	39
1.3 Types de Système Tutoriel Intelligent . . . . .	41
1.4 Évaluation des systèmes d'apprentissage en ligne en terme de performance pédagogique . . . . .	44
1.5 Évaluation des systèmes d'apprentissage en ligne en terme de fonctionnalités	48
1.6 Exemples de quelques systèmes tutoriels intelligents . . . . .	56
1.7 L'exercice dans les plateformes de formation classique et dans les STI . . . .	59
1.8 Discussion et conclusion . . . . .	62
<b>Chapitre 2 Évaluation de l'apprenant pour l'adaptation pédagogique</b>	<b>64</b>
2.1 Introduction . . . . .	64
2.2 Méthodes d'évaluation automatisées des apprenants . . . . .	65
2.3 Évaluation dans les Systèmes Tutoriels Intelligents . . . . .	67
2.4 Évaluations adaptées . . . . .	68
2.5 Systèmes d'adaptation dans les plateformes d'apprentissage en ligne . . . .	70
2.6 Adaptation dans les Systèmes Tutoriels Intelligents . . . . .	74
2.7 Bilan . . . . .	76
2.8 Conclusion . . . . .	78

---

Le e-learning, en tant que segment important de l'environnement d'enseignement, représente une occasion unique d'apprendre de façon indépendante, quels que soient l'heure et le lieu pour acquérir des connaissances selon les principes de l'apprentissage traditionnelle. Le E-learning offre un certain nombre d'avantages pour les personnes et les entreprises qui souhaitent élaborer un programme scolaire ou contenu pédagogique [Bernardin, 2007]. Si les apprenants sont tous ensemble dans une classe ou dispersés ou dans différents fuseaux horaires, ils peuvent encore utiliser le même document de cours à des moments qui leur conviennent.

Dans cette partie du manuscrit, nous nous intéressons aux systèmes tutoriels intelligents (STI) en tant que classe particulière des systèmes d'e-learning, qui soutiennent et améliorent l'apprentissage et l'enseignement des connaissances d'un domaine spécifique. Ces systèmes interactifs peuvent exécuter des situations d'apprentissage dans un contexte donné. Dans ces systèmes, les apprenants ont des préférences, des besoins et des approches différentes en matière d'apprentissage.

Dans cette partie, nous abordons l'historique des systèmes tutoriels intelligents (STI), les travaux existants pour l'évaluation de l'apprenant et l'adaptation de la ressource et de la stratégie d'apprentissage dans ces systèmes. Dans le chapitre 1, nous présentons la définition des systèmes tutoriels intelligents, leur classification, leur architecture principale et leurs différents modèles. Une revue de quelques systèmes STI dans différents domaines d'application est aussi mentionnée dans ce chapitre. Le chapitre 2 détaille les approches d'évaluations et les systèmes d'adaptation existants dans les systèmes d'e-Learning et dans les systèmes tutoriels intelligents (STI). Ce chapitre est consacré aussi à l'étude des systèmes d'adaptation intégrés dans les systèmes d'apprentissage en ligne et particulièrement les systèmes tutoriels intelligents. Ainsi, nous avons identifié les verrous scientifiques pour lesquels nous avons développé nos solutions présentées dans la suite du manuscrit.

# Chapitre 1

## Systemes Tutoriels Intelligents

### 1.1 Introduction

Dans ce chapitre, nous nous intéressons à une classe de systèmes d'e-Learning appelée systèmes tutoriels intelligents (STI). La motivation derrière le choix d'étude de ces systèmes revient au fait qu'ils répondent aux exigences des utilisateurs en termes de diversification des activités qu'ils dispensent. Comme, par exemple, des cours, des travaux dirigés, des travaux pratiques, etc., qui permettent une formation individualisée un-à-un en imitant les activités des enseignants en présentiel. La structuration des activités est assurée par la forte interaction entre le modèle d'expert et le modèle pédagogique. Ils remplissent aussi les conditions nécessaires pour le fondement d'une évaluation adaptée de l'apprenant qui nécessite une connaissance approfondie de son modèle, son historique, son évolution, etc.

Les STI permettent d'identifier le niveau de connaissances des apprenants et d'utiliser des stratégies d'apprentissage pour améliorer leurs niveaux de connaissances. Ils sont destinés à soutenir et à améliorer le processus d'enseignement et d'apprentissage dans le domaine de connaissances choisi tout en respectant l'individualité de l'apprenant.

Les STI sont fondés sur des principes tels que :

1. les connaissances du domaine ;
2. les connaissances sur la pédagogie ;
3. les méthodes permettant d'appliquer ces principes et la connaissance sur les méthodes et techniques de modélisation de l'apprenant pour acquérir des compétences.

Nous présentons dans ce chapitre les concepts définissant les Systèmes Tutoriels Intelligents (STI) ainsi que leurs caractéristiques. Ces systèmes sont classés en catégories suivant les outils de tutorat qui les constituent. Toutefois, les STI partagent un noyau commun basé sur trois



modèles, à savoir : le modèle expert, le modèle apprenant, le modèle pédagogique. Ces modèles nous fournissent les caractéristiques et paramètres nécessaires pour connaître l'état et le niveau d'acquisition des connaissances de l'apprenant. Enfin, dans le bilan du chapitre, nous avons comparé les systèmes STI en termes de fonctionnalités proposées et nous avons identifié le type de STI dans lequel nous avons développé nos approches d'évaluation des apprenants et l'adaptation individualisée des contenus et des stratégies pédagogiques.

## 1.2 Définitions et caractéristiques des systèmes tutoriels intelligents

Un Système Tutoriel Intelligent (STI) est conçu pour simuler un comportement des tuteurs humains et orienter les apprenants [Koedinger et Tanner, 2013]. Les apprenants peuvent l'utiliser pour poser des questions, analyser la liste des réponses et s'assurer un enseignement personnalisé et une rétroaction efficace. Les STI permettent une formation en mode face-à-face en imitant les activités des enseignants humains (en termes de quoi, quand et comment enseigner un contenu pédagogique défini [Etienne, 1987; Ohlsson, 1987]). Ce type de systèmes peut modifier son comportement de tutorat en temps réel en fonction du profil de l'apprenant ou ajuster sa base de connaissances pour des interactions plus efficaces avec les apprenants. ans [Poole et Mackworth, 2010], un Système Tutoriel Intelligent a comme entrées les paramètres suivants :

- connaissances préalables fournies par le concepteur sur le sujet enseigné, les stratégies d'enseignement, les erreurs possibles, et les incompréhensions des apprenants ;
- l'expérience passée que le système de tutorat a acquise en interagissant avec les apprenants, sur leurs erreurs et le nombre d'exemples nécessaires pour inculquer un concept donné. Cela peut être des informations sur les apprenants en général ou au sujet d'un apprenant en particulier ;
- l'importance de chaque activité et le niveau de réussite de l'apprenant ;
- observations des résultats de tests et d'interactions (ou non interactions) de l'étudiant avec le système ;
- Les apprenants peuvent également travailler en collaboration, par exemple poser des questions ou fournir de nouveaux exemples.

La sortie du système tutoriel est l'information présentée et affichée à l'étudiant, les tests que les étudiants devraient subir, les réponses aux questions, et les rapports envoyés aux tuteurs et/ou aux enseignants. Un système STI est caractérisé par les aspects suivants :

- une décomposition hiérarchique des tâches d'enseignement. Les apprenants doivent acquérir les compétences et les connaissances d'un niveau débutant avant celles du niveau supérieur. Le système tutoriel a des stratégies d'enseignement composées de briques de formation qui sont détaillées et découpées en composantes fines ;
- un système tutoriel est capable de raisonner en termes de l'état de l'apprenant. Cependant, il est nécessaire d'avoir de multiples sources d'information pour le domaine de l'apprenant et de celui de la formation ;
- l'horizon de planification est statique parce que l'apprenant ne peut pas exécuter simultanément deux activités différentes. Deux modes de scénarisation existent. Dans un horizon défini, le scénario d'activité ou sous-activité est prédéfini par un concepteur/formateur, par exemple une séquence d'activités peut être constituée d'un cours d'apprentissage, suivi par des travaux dirigés et par un test. Dans l'horizon indéfini, le système ne peut pas savoir le nombre d'étapes nécessaires pour qu'un étudiant maîtrise un concept donné ;
- l'incertitude joue un rôle important. Le système ne peut pas observer directement les connaissances de l'apprenant. Les entrées sont des éléments de détection, basées sur les questions posées par l'apprenant, et les résultats des tests. Le système ne saura pas avec certitude l'effet d'un scénario d'enseignement particulier sur le niveau d'acquisition de connaissances par l'apprenant ;
- un objectif simple est défini pour l'enseignement d'un concept ou d'une activité particulière ;
- Les préférences des apprenants sont aussi prises en compte car elles sont nécessaires pour personnaliser l'enseignement ou l'activité ;
- le système est en mesure d'apprendre davantage sur le fonctionnement de la stratégie d'enseignement, le déroulement des questions vis-à-vis des concepts de test. Le système pourrait aussi apprendre des connaissances générales ou particulières concernant un sujet donné, par exemple sur les mécanismes d'enseignement ou sur les connaissances maîtrisées ou pas par l'apprenant.

En général, les aspects fondamentaux d'un système de tutorat ne dépendent pas seulement du contexte d'utilisation ou de population apprenante, mais aussi des fonctionnalités du domaine que le système possède indépendamment ainsi que du type de modèles qui le compose.

Les chercheurs dans le domaine des STI ont souligné la différence entre la théorie des systèmes intelligents et celle des systèmes d'enseignement classique. Par ailleurs, ils ont articulé une structure conceptuelle flexible à quatre composantes pour les STI, même si les STI eux-mêmes varient considérablement dans leur conception. Les quatre composantes conceptuelles généralement acceptées [Sottolare *et al.*, 2013] sont les suivantes :

- une interface qui communique avec l'apprenant en présentant et en recevant de l'information. Souvent contrainte au domaine (programmation, par exemple), l'interface détermine les mouvements que l'apprenant peut effectuer pour résoudre des problèmes, rechercher des informations ou répondre à des questions ;
- un modèle de domaine qui représente les connaissances que l'apprenant est censé apprendre. Le modèle est un ensemble de propositions logiques, de règles de production, d'énoncés en langage naturel ou de tout autre format de représentation des connaissances appropriées ;
- un modèle de l'apprenant qui représente les aspects pertinents des connaissances de l'apprenant, déterminés par ses réponses aux questions ou autres interactions avec l'interface. Bien que ce modèle puisse être un sous-ensemble ou une superposition du modèle de domaine, dans certains STI, le modèle de l'apprenant représente des méconnaissances communes ou d'autres « Bagues » dans ses connaissances ;
- un modèle de tuteur qui représente des stratégies d'enseignement telles que l'offre d'un indice lorsque l'apprenant est incapable de générer une réponse correcte ou assigner un problème qui nécessite des connaissances qui sont légèrement au-delà du modèle actuel de l'apprenant.

Il est à noter que les STI n'ont pas tous les quatre composantes architecturales [Adenowo et Adenowo, 2016]. Par exemple, un STI peut avoir une base de connaissances unique qui sert de modèle de domaine et de modèle d'apprenant et qui n'a pas de représentation explicite de la stratégie d'enseignement. Nous considérons dans notre cas, le processus de modélisation des apprenants et son utilisation pour adapter l'instruction et qui constitue la caractéristique essentielle qui distingue les STI des autres systèmes d'enseignement informatisés.

## 1.3 Types de Système Tutoriel Intelligent

Étant donné que la modélisation des apprenants est un élément essentiel de la conception des STI et qu'il est utile d'analyser les résultats d'apprentissage associés aux types les plus courants de modélisation des apprenants, les STI sont catégorisés dans les quatre types suivants :

1.3.1 - STI basés sur les attentes des méconnaissances

1.3.2 - STI basés sur le modèle de traçage

1.3.3 - STI basés sur la modélisation par les contraintes

1.3.4 - STI basés sur la modélisation en réseau bayésien

### 1.3.1 STI basés sur les attentes des méconnaissances

Les dialogues de tutorat dans lesquels le système et l'apprenant échangent des idées en langage naturel ont été un objectif ambitieux pour la recherche sur les STI depuis SCHOLAR [Carbonell, 1970]. « AutoTutor » est un STI qui implémente un dialogue en langage naturel pour l'enseignement qui, comme beaucoup de tutorat humain, implique un contenu verbal imprécis, un niveau de connaissance faible à moyen des utilisateurs [Graesser *et al.*, 2004]. « AutoTutor » modélise les connaissances des apprenants en faisant correspondre leurs réponses aux passages de texte représentant les attentes (c'est-à-dire les objectifs d'apprentissage) et les méconnaissances anticipées dans le domaine. Les attentes et les méconnaissances précisées constituent un modèle de domaine multidimensionnel. L'appariement est effectué par une méthode statistique appelée analyse sémantique latente qui mesure une similarité entre une agrégation des réponses de l'apprenant et chaque attente et méconnaissance dans le modèle de domaine [Laudauer *et al.*, 1998]. « AutoTutor » utilise le résultat du processus d'appariement pour générer des « mouvements de dialogue » scriptés tels que conseils, rétroactions, instructions et assertions [Graesser *et al.*, 2004]. Dans l'enseignement de la physique qualitative, « Why2-Atlas » sur les attentes des méconnaissances comme « AutoTutor » pour analyser les essais des apprenants. La principale différence est que « AutoTutor » utilise une technique statistique pour l'appariement, tandis que « Why2-Atlas » utilise des analyses symboliques [VanLehn *et al.*, 2002]. Dans le domaine de la médecine, « CIRCSIM-Tutor » un système de tutorat intelligent qui se base aussi sur les méconnaissances des apprenant en fournissant des conseils qui peuvent être invoquées dans différents contextes. Sinon, le système peut donner un indice physiologique [Glass, 2001].

### 1.3.2 STI basés sur le modèle de traçage

Dans la conception d'un tuteur cognitif, la compétence à apprendre est modélisée par un ensemble de règles de production qui peuvent être activées pour résoudre automatiquement les problèmes dans le domaine [Anderson *et al.*, 1995; Roll *et al.*, 2004; Anderson et Lebiere, 2014]. Les règles de production, qui consistent en des opérations et des conditions sans lesquelles elles sont déclenchées, sont sélectionnées pour former une émulation psychologiquement réaliste de la façon dont les humains résolvent les problèmes dans le domaine. Les opérations dans le modèle de domaine sont exposées dans l'interface où l'apprenant peut les sélectionner pour progresser vers une solution au problème. Lorsque l'apprenant sélectionne les opérations, un processus de suivi des modèles les associe à une série de règles de production dans le modèle de domaine. Si une erreur est détectée, l'apprenant reçoit une rétroaction immédiate et peut choisir une autre opération. Une fois que l'utilisation d'une règle de production par l'apprenant est identifiée par le modèle de traçage, une procédure bayésienne appelée « traçage des connaissances » peut être utilisée pour mettre à jour une estimation de la probabilité

que la connaissance soit correctement apprise. Ainsi, le modèle de l'apprenant multidimensionnel dans un tuteur cognitif de traçage des connaissances est constitué comme des probabilités assignées aux règles de production dans le modèle de domaine.

### 1.3.3 STI basés sur la modélisation par les contraintes

La modélisation par les contraintes est une technique établie pour la modélisation des apprenants qui est fondamentalement différente du modèle de traçage des connaissances utilisées chez les tuteurs cognitifs [Kodaganallur *et al.*, 2005; Suraweera et Mitrovic, 2002]. Basée sur la théorie de l'apprentissage de performance des erreurs, la modélisation par les contraintes représente les connaissances du domaine comme des contraintes logiques en reliant chaque contrainte aux états qui pourraient survenir dans la résolution d'un problème [Greer et McCalla, 2013]. Une contrainte se compose de trois éléments : (a) une condition de pertinence qui indique quand la contrainte est applicable, (b) une condition de satisfaction qui teste l'état actuel de la solution de l'apprenant, et (c) un message de rétroaction lorsque l'état de la solution échoue à la condition de satisfaction, informe l'apprenant de l'erreur et lui rappelle le principe qui a été violé par l'erreur [Mitrovic *et al.*, 2007]. Si aucune contrainte n'est violée, l'apprenant est considéré comme étant sur le bon chemin. Pour la modélisation des connaissances du domaine, les contraintes jouent un rôle similaire dans la modélisation basée sur les règles de production dans le modèle de traçage, sauf que les contraintes ne peuvent pas être exécutées pour générer une solution au problème donné. Les contraintes peuvent être utilisées pour modéliser les connaissances des apprenants de différentes façons. Par exemple, un STI peut représenter les connaissances d'un apprenant en tant que contraintes jugées pertinentes, satisfaites ou violées pendant une session de résolution de problèmes.

### 1.3.4 STI basés sur la modélisation en réseau bayésien

Un réseau bayésien est un outil de raisonnement probabiliste et de représentation de connaissances incertaines [Pearl, 2014; Conati et Zhao, 2004]. Dans un STI, les réseaux bayésiens sont utilisés pour représenter un modèle de domaine multidimensionnel constitué de plusieurs variables. Les connexions entre les variables sont spécifiées pour former un réseau, et l'inférence sur la valeur d'une variable dans le réseau est réalisée par des calculs bayésiens sur d'autres variables qui lui sont connectées [Millán *et al.*, 2010]. Par exemple, une liste de variables binaires cibles représentant des concepts et des méconnaissances peuvent constituer le modèle de domaine et une liste de variables de preuve représentant les éléments de test peuvent être connectées aux variables cibles. En prenant comme point de départ la performance de l'apprenant sur les items du test, le réseau bayésien calcule la probabilité d'acquisition que l'apprenant ait obtenu pour chaque concept et chaque méconnaissance. Les réseaux bayésiens peuvent être

utilisés pour créer des modèles beaucoup plus complexes, tels que les modèles dynamiques de résolution de problèmes chez les tuteurs qui fournissent des conseils et un encadrement d'apprentissage [Conati *et al.*, 2002]. Le réseau bayésien est une méthode flexible qui peut être utilisée pour mettre en œuvre différents types de modèles d'apprenant, y compris des aspects de la modélisation basée sur les contraintes et du traçage des connaissances.

## 1.4 Évaluation des systèmes d'apprentissage en ligne en terme de performance pédagogique

Dans [Murray, 1993, 1999], de nombreuses approches d'évaluation des STI décrites continuent d'être utilisées, de nouvelles approches pour évaluer les tuteurs intelligents et autres environnements d'apprentissage adaptatifs continuent d'évoluer, en réponse à l'augmentation des ressources en termes de stockage et d'utilisation. Les approches prometteuses pour l'évaluation des environnements d'apprentissage sont caractérisées par :

1. l'utilisation du web ouvert et l'approvisionnement par la foule (crowd sourcing) pour évaluer les systèmes tutoriels intelligents ;
2. comparaison des décisions prises par des environnements d'apprentissage adaptatifs et des experts humains ;
3. utilisation des apprenants simulés dans l'évaluation des environnements d'apprentissage ;
4. élimination des biais de sélection tout en évaluant les interventions éducatives ;
5. examen des trajectoires d'apprentissage pour évaluer les systèmes d'apprentissage ;
6. évaluations dérivées de méga-quantités des données d'apprentissage.

Nous décrivons brièvement ci-dessous chacune de ces six caractéristiques.

- **Le savoir (ou non) des foules** : alimenté en partie par le développement des plateformes comme Amazon's Mechanical Turk, l'approvisionnement par la foule « crowd sourcing » est utilisé et étudié par le monde des affaires, la communauté de jeux et d'interaction homme-machine (IHM). Les types d'approvisionnement par la foule peuvent être des concours, des communautés de collaboration, etc. [Boudreau et Lakhani, 2013]. Avec des incitations intelligentes impliquant du plaisir, de l'altruisme et de l'argent, de telles approches peuvent tirer parti d'un échantillon potentiellement énorme d'utilisateurs qui peuvent aider à étudier un système d'apprentissage et à donner des retours d'informations

détaillés. En appliquant l'approvisionnement par la foule à l'évaluation des systèmes éducatifs, il existe des problèmes d'adéquation des volontaires aux objectifs d'enseignement et d'apprentissage du système, la véracité des réactions et des réponses, et le sérieux avec lequel les volontaires interagissent avec le système. Les auteurs dans [Kittur *et al.*, 2008] examinent la faisabilité de l'évaluation participative avec « Mechanical Turk », notant son potentiel de collecte de mesures d'utilisateurs pour des études formatives avec des techniques telles que les enquêtes, le prototypage rapide, et les mesures de performance quantitatives. Ils signalent également le danger de l'utilisation des systèmes, avec des résultats désastreux, si les tâches ne sont pas soigneusement élaborées. L'approvisionnement par la foule est probablement le moyen le plus utile de rassembler des données formatives auprès d'une large base d'utilisateurs et d'obtenir une vérification rapide de la réalité d'un système avant d'investir trop d'efforts dans une étude plus contrôlée. L'approvisionnement par la foule peut ne pas être un moyen fiable ou valide pour évaluer un système d'apprentissage.

- **Mesures Kappa et d'experts humains** : Cette méthode permet de comparer les décisions ou les choix d'un système intelligent automatisé à ceux d'experts humains, ce type d'évaluation basée sur des critères peut aider à déterminer si le système intelligent conforme à ce qu'un expert pourrait faire. Les mesures kappa sont d'une estimation consensuelle de la fiabilité inter-évaluateur. Ils sont rarement utilisés pour des tests de signification statistique. Les mesures Kappa donnent plutôt une indication d'un accord meilleur ou moins bon entre ou parmi les décideurs [Stemler, 2004].

Plusieurs chercheurs de la communauté AIED (Artificial Intelligence in Education) ont utilisé Kappa dans l'évaluation des comportements des systèmes intelligents [Rourke *et al.*, 2001]. L'utilisation de Kappa dans l'évaluation des environnements d'apprentissage adaptatifs vise généralement à démontrer que les décisions du système correspondent à ce qu'un expert humain ferait. Il y a deux problèmes fondamentaux avec de telles comparaisons. Quand il s'agit de décisions ou de choix d'actions éducatives, en particulier là où la subtilité du contexte joue un rôle, l'accord entre les experts humains est souvent assez faible. Cela donne une grande latitude pour les décisions ou les choix d'un système d'apprentissage en ligne, tout en restant accord avec les experts humains. Le deuxième problème est l'« l'étalon-or » où le système d'apprentissage en ligne soit de loin supérieur à tout expert humain dans ses jugements ou ses décisions : les mesures d'accord Kappa seraient assez faibles, même si le système fonctionne exceptionnellement bien [Viera *et al.*, 2005]. Lorsque l'on compare un système avec un expert humain, une bonne mesure Kappa nous informe que le système ne se comporte ni bien pire ni beaucoup mieux qu'un expert humain.

- **Apprenants simulés** : Le biais et la petite taille de l'échantillon continuent d'être des problèmes pour les évaluations expérimentales des environnements d'apprentissage. Même

si des volontaires indépendants sont recrutés comme sujets expérimentaux, ils peuvent être peu nombreux et ne pas refléter la population cible à laquelle le système est destiné. Frustrés par des tailles d'échantillon petites ou non représentatives, certains chercheurs se sont tournés vers des sujets synthétiques, c'est-à-dire des apprenants simulés. Les apprenants simulés sont des systèmes logiciels ou des agents qui peuvent être conçus pour refléter la population des apprenants prévus et être mis à l'échelle pour fournir une taille d'échantillon aussi grande que souhaitée. Cependant, un biais peut être introduit en surimposant l'apprenant simulé à l'environnement d'apprentissage, ce qui peut conduire à des prédictions d'apprentissage irréalistes. Les apprenants simulés peuvent aider à l'évaluation formative des environnements d'apprentissage ou à la validation de composantes spécifiques d'un système, mais les tests avec des apprenants simulés ne peuvent remplacer une évaluation sommative complète et nécessaire à l'étude de l'impact éducatif [Matsuda *et al.*, 2015]. Il y a toujours la possibilité que dans une situation réelle, les apprenants et le système se comportent de manière imprévue par l'expérimentateur.

- **Nettoyer le biais de sélection** : Un biais de sélection peut survenir lorsque l'on se fie à des sujets volontaires dans le cadre des études expérimentales formelles. Certaines personnes sont plus susceptibles de faire du bénévolat que d'autres et on ne peut être certain que les bénévoles soient représentatifs de la population cible. Ce genre de problème se produit dans les milieux de recherche où un sous-ensemble d'apprenants décide de se porter volontaires pour un traitement expérimental, comme l'utilisation d'un environnement d'apprentissage en ligne. À moins que suffisamment de volontaires se soient manifestés pour mettre en place un groupe de contrôle randomisé et un groupe de traitement, les non-volontaires deviennent un contrôle de substitution. Des problèmes similaires peuvent se produire lorsqu'une section de cours utilise un environnement d'apprentissage tandis qu'une autre section du même cours agit comme un groupe de contrôle. Les différences entre les deux groupes peuvent refléter des variations autres que l'environnement d'apprentissage.

L'appariement par score de propension est un type d'étude non randomisée qui peut être utilisé pour minimiser le biais de sélection et estimer les effets des traitements sur les résultats. Il fonctionne en faisant correspondre les apprenants du groupe de traitement avec les groupes de comparaison. Seuls les apprenants présentant un haut degré de similarité sur les variables pertinentes devraient être jumelés [Austin, 2011]. On suppose que pour bien fonctionner, le groupe de comparaison doit être beaucoup plus grand que le groupe de traitement, de sorte qu'un ensemble représentatif de comparateurs puisse être identifié comme étant contrôleurs. Cette méthode prometteuse n'a été appliquée que récemment dans l'évaluation des environnements d'apprentissage, en particulier dans l'étude des MOOC, où la disponibilité de grands groupes de sujets et les progrès du big data et de



l'apprentissage en ligne permettent de détecter des sujets comparables [Brooks *et al.*, 2015].

- **Utilisation des trajectoires d'apprentissage** : Mesurer les changements dans l'apprentissage en raison de certaines interventions éducatives est très difficile. Les mesures standards des gains d'apprentissage dans les mesures pré-post lors de la comparaison de certains traitements et contrôles sont rarement significatives. Un moyen plus subtil pour mesurer les gains d'apprentissage peut être obtenu en analysant les trajectoires d'apprentissage. Provenant de l'apprentissage en ligne, les trajectoires d'apprentissage représentent les taux d'erreur dans les performances des apprenants par rapport à une estimation du niveau de connaissances des apprenants. Cela peut être utilisé pour déterminer si de nouvelles modifications améliorent ou dégradent les performances de l'apprenant. Une illustration de la méthodologie a été effectuée en utilisant les trajectoires d'apprentissage dans l'évaluation de systèmes éducatifs adaptatifs et dans les études formatives et sommatives [Martin *et al.*, 2005, 2011].

- **Exploration de données éducatives et d'apprentissage** :

De nombreux systèmes éducatifs rassemblent des données de parcours détaillées des actions de l'apprenant. De même, les logs vidéos, les données de suivi de l'œil (eye-tracking), les fichiers journaux des LMS et les plateformes MOOC accumulent d'énormes quantités de données. Tenter de donner un sens à des données de logs très pose un défi technique important pour l'exploration des données. Des modèles ou des caractéristiques significatives peuvent parfois être identifiées à l'avance grâce à une compréhension a priori du domaine de l'activité de l'apprenant, ou à l'aide des machines d'apprentissage et techniques de filtrage par motifs. L'agrégation des données fines en modèles significatifs nécessite un processus de fusion et d'ingénierie de données ainsi qu'une bonne compréhension des méthodes statistiques et d'exploration de données. Les outils de visualisation de données peuvent également être utiles pour reconnaître des caractéristiques qui n'ont peut-être pas été prévues à l'avance. Ces techniques peuvent avoir des applications dans la modélisation et le profilage d'utilisateurs, la modélisation de domaine, et l'étude de composants d'apprentissage et de principes pédagogiques [Mining, 2012]. L'application de ces techniques d'exploration de données d'apprentissage et d'analyse à de grandes quantités de données des apprenants à granularité fine est probablement le domaine qui croît le plus activement dans l'évaluation des environnements d'apprentissage [Baker et Inventado, 2014].

La variété des systèmes tutoriels intelligents en termes de fonctionnalité rend le choix d'un système difficile. La majorité des apprenants ont besoin d'apprendre dans un STI qui répond à leurs attentes et objectifs, par exemple, un système qui offre une interaction en langage naturel,

gère les méconnaissances, fournit des conseils et rétroactions, donne des instruction et asser-tions en assistant l'apprenant. D'autres cherchent des STIs basés sur des règles de production et des modèles de traçage de connaissances ou de modélisation par contraintes qui aident et assistent la résolution des problèmes de la même manière qu'un humain pourrait le faire. Une classification est utile pour les enseignants et les apprenants afin qu'ils puissent identifier le système STI qui réponde à leurs besoins et objectifs en termes de fonctionnalités. L'évaluation des systèmes d'apprentissage en ligne en terme de fonctionnalité est décrite dans la section suivante.

## **1.5 Évaluation des systèmes d'apprentissage en ligne en terme de fonctionnalités**

Il existe peu de méthodes dans la littérature qui évaluent les plateformes d'apprentissage en ligne en termes de critères. [Stufflebeam, 1994] présente l'approche la plus utilisée « Qualitative Weight and Sum », désignée par QWS qui s'appuie sur une liste de critères pondérés pour l'évaluation des systèmes e-Learning [Graf et List, 2005], et la méthode AHP (Analytic Hierarchy Process) utilisée pour traiter des processus décisionnels complexes en traduisant les symboles définis dans QWS en valeurs numériques [Hamtini et Fakhouri, 2012].

Comme ces méthodes renvoient des valeurs numériques ou des moyennes [Graf et List, 2005], elles peuvent être moins expressives et non intuitives du point de vue de l'utilisateur pour l'évaluation de la qualité et le classement des système d'apprentissage en ligne. À cet effet, nous proposons une approche hybride pour l'évaluation du système et classement combinant les valeurs QWS, les relations de préférence symboliques et les opérateurs de comparaison formels, qui se sont révélés être des ordres totaux permettant de distinguer les plateformes e-Learning optimales du point de vue de l'utilisateur [Chachoua *et al.*, 2016b; Chachoua et Tamani, 2016].

Dans cette section, nous détaillons notre approche pour l'évaluation et le classement des plateformes d'e-Learning en se reposant sur des symboles empruntés à la méthode QWS [Stufflebeam, 1994], des relations de préférence qualitative et des opérateurs de comparaison. Dans la sous-section 1.5.1, nous présentons notre approche d'évaluation et dans la sous-section 1.5.2, nous montrons l'utilisation de notre approche pour le classement des plateformes d'e-learning.

### **1.5.1 Approche symbolique pour l'évaluation des plateformes d'e-learning**

Nous définissons les symboles d'évaluation comme suit.

**Définition 1.5.1** (Symboles d'évaluation). Les symboles d'évaluation tels que définis dans l'approche QWS sont :

$E = \text{essentiel}$ ,  $* = \text{extrêmement important}$ ,  $\# = \text{très important}$ ,  $+ = \text{important}$ ,  $| = \text{marginale-ment important}$  et  $0 = \text{pas important}$ . Nous désignons par  $\mathcal{S} = \{E, *, \#, +, |, 0\}$  un ensemble ordonné de symboles d'évaluation.

Nous définissons une relation de préférence *plus préféré ou égal à*, notée  $\succeq$ , et *moins préféré ou égal à*, notée  $\preceq$ , sur l'ensemble de symboles d'évaluation  $\mathcal{S}$  comme suit.

**Définition 1.5.2** (Relations de préférence  $\succeq$  et  $\preceq$ ). Considérons  $\mathcal{S} = \{E, *, \#, +, |, 0\}$  un ensemble ordonné de symboles d'évaluation tels que :

- Position 1 est pour le symbole  $E$ , notée par  $pos_{\mathcal{S}}(E)$
- Position 2 est pour le symbole  $*$ , notée par  $pos_{\mathcal{S}}(*)$
- Position 3 est pour le symbole  $\#$ , notée par  $pos_{\mathcal{S}}(\#)$
- Position 4 est pour le symbole  $+$ , notée par  $pos_{\mathcal{S}}(+)$
- Position 5 est pour le symbole  $|$ , notée par  $pos_{\mathcal{S}}(|)$
- Position 6 est pour le symbole  $0$ , notée par  $pos_{\mathcal{S}}(0)$

Nous définissons la relation de préférence *plus préférée ou égale à*  $\succeq$  sur  $\mathcal{S}$  comme suit.

$$\forall (a, b) \in \mathcal{S}^2 : a \succeq b \text{ ssi } pos_{\mathcal{S}}(a) \leq pos_{\mathcal{S}}(b) \quad (1.1)$$

La relation de préférence *moins préférée ou égale à*, notée par  $\preceq$ , est définie comme suit.

$$\forall (a, b) \in \mathcal{S}^2 : a \preceq b \text{ ssi } pos_{\mathcal{S}}(a) \geq pos_{\mathcal{S}}(b) \quad (1.2)$$

Nous pouvons facilement prouver que la relation de préférence  $\succeq$  est un ordre total.

**Propriété 1.** (*Propriétés de l'ordre total*). Les relations de préférence  $\succeq$  et  $\preceq$  sont deux ordres totaux.

Sur la base des relations de préférences définies ci-dessus, nous définissons deux opérateurs de comparaison appelés *prefMin* et *prefMax*, de sorte qu'il sera possible de comparer les systèmes sur chaque critère décrivant ces derniers. Ces opérateurs servent de moyen d'agréger les évaluations obtenues pour les critères du système.

**Définition 1.5.3.** (Opérateurs de comparaison *prefMax* et *prefMin*). Les opérateurs *prefMax* et *prefMin* sont définis par les formules (1.3) et (1.4) respectivement.

La fonction *prefMax* est définie par la formule (1.3).

$$\begin{aligned}
 \mathcal{S} \times \mathcal{S} &\rightarrow \mathcal{S} \\
 (a, b) &\mapsto \max(a, b) = \begin{cases} a & \text{si } (a \succeq b) \\ b & \text{sinon.} \end{cases} \tag{1.3}
 \end{aligned}$$

La fonction *prefMin* est définie par la formule (1.4).

$$\begin{aligned}
 \mathcal{S} \times \mathcal{S} &\rightarrow \mathcal{S} \\
 (a, b) &\mapsto \min(a, b) = \begin{cases} a & \text{si } (a \preceq b) \\ b & \text{sinon.} \end{cases} \tag{1.4}
 \end{aligned}$$

Lorsque nous appliquons les opérateurs de comparaison *prefMax* et *prefMin* sur notre ensemble symbolique  $\mathcal{S}$ , nous obtenons la Table 1.1.

**Propriété 2.** (les propriétés de *prefMax*). *prefMax* est associatif, commutatif, idempotent, il a  $E$  comme élément absorbant et  $0$  comme élément neutre.

**Propriété 3.** (les propriétés de *prefMin*). *prefMin* est associatif, commutatif, idempotent, il a  $0$  comme élément absorbant et  $E$  comme élément neutre.

TABLE 1.1 – La table des opérateurs *prefMax* et *prefMin*.

<i>prefMax</i>	$E$	*	#	+		0
$E$	$E$	$E$	$E$	$E$	$E$	$E$
*	$E$	*	*	*	*	*
#	$E$	*	#	#	#	#
+	$E$	*	#	+	+	+
	$E$	*	#	+		
0	$E$	*	#	+		0

<i>prefMin</i>	$E$	*	#	+		0
$E$	$E$	*	#	+		0
*	*	*	#	+		0
#	#	#	#	+		0
+	+	+	+	+		0
						0
0	0	0	0	0	0	0

### 1.5.2 Utilisation de nos opérateurs de comparaison pour classer les systèmes d'e-Learning

L'évaluation des plateformes d'e-Learning repose sur des catégories, chacune définissant certains critères [Atthirawong et MacCarthy, 2002], par exemple la catégorie *Outils de Com-*

*munication* et leur critère tel que *Chat*. Les cinq catégories considérées dans l'évaluation de la plateforme sont les suivantes :

- Outils de communication
- Logiciel et installation
- Outils administratifs et sécurité
- Outils de présentation matériels
- Caractéristiques de gestion

Pour évaluer chaque catégorie, nous utilisons les opérateurs de comparaison *prefMax* et *prefMin*. Mais pour évaluer un système e-Learning considéré, nous avons besoin de l'évaluation des cinq catégories. Pour cela, nous définissons deux opérateurs d'agrégation appelés *prefMinMax* et *prefMaxMin*, qui sont basés sur nos opérateurs de comparaison.

**Définition 1.5.4.** (*prefMinMax*). Considérons  $\mathcal{A}$  une matrice de  $n$  lignes et  $m$  colonnes de symboles d'évaluation de  $\mathcal{S}$ . Nous définissons la valeur de satisfaction minimale garantie comme suit.

Nous désignons une matrice de  $\mathcal{A}$  comme :

$$A = (a_{ij})_{\substack{1 \leq i \leq m \\ 1 \leq j \leq n}} \text{ et } a_{ij} \in \mathcal{S}$$

Nous définissons *prefMinMax* de  $\mathcal{A}$  comme :

$$\begin{aligned} \mathcal{S}^{m \times n} &\rightarrow \mathcal{S} \\ A &\mapsto \text{prefMinMax}(A) = \text{prefMin}_{1 \leq i \leq m}(\text{prefMax}_{1 \leq j \leq n}(a_{ij})) \end{aligned} \tag{1.5}$$

**Définition 1.5.5.** (*prefMaxMin*). Nous définissons la valeur de satisfaction maximale possible de  $\mathcal{S}^{m \times n}$  comme *prefMaxMin* :

$$\begin{aligned} \mathcal{S}^{m \times n} &\rightarrow \mathcal{S} \\ A &\mapsto \text{prefMaxMin}(A) = \text{prefMax}_{1 \leq i \leq m}(\text{prefMin}_{1 \leq j \leq n}(a_{ij})) \end{aligned} \tag{1.6}$$

### 1.5.3 Exemple illustratif

Nous appliquons notre approche d'évaluation des systèmes e-Learning à un ensemble de neuf plateformes d'e-learning open source énumérées ci-dessous et qui ont été testées et comparées.

1. Claroline : version 1.9.2, <http://www.claroline.net>
2. Dokeos : version 2.1.1, <http://www.dokeos.com/fr>
3. eFront : version 3.6.11, <http://www.efrontlearning.net>
4. ILIAS : version 4.1.3, <http://www.ilias.de>
5. Open ELMS : version 7, <http://www.openelms.org>
6. Ganesha : version 4.5, <http://www.ganesha.fr>
7. Olat : version 7.2.1, <http://www.olat.org>
8. AnaXagora : version 3.5, <http://www.anaxagora.tudor.lu>
9. Sakai : version 10.4, <https://sakaiproject.org>

Chaque critère prend une valeur symbolique à partir de l'ensemble  $S$  en fonction des opinions des utilisateurs. Pour obtenir l'évaluation de chaque critère, nous avons effectué des sondages dans notre université impliquant des étudiants en graduation (petit groupe de 10 étudiants), qui ont testé chaque plateforme d'e-learning pendant une séance d'entraînement (2 heures). Nous sommes conscients que le processus est subjectif et qu'un autre groupe d'étudiants ou d'utilisateurs peut exprimer des opinions différentes sur les plateformes d'e-Learning. Nous rappelons que cette collecte de données vise à illustrer l'utilisation de notre approche. Les valeurs obtenues pour chaque critère dans sa catégorie sont résumées dans la Table 1.2. L'application de notre approche sur l'ensemble des systèmes considérés est effectuée comme suit.

1. pour chaque catégorie dans la Table ??, nous calculons les valeurs de  $prefMin$  et  $prefMax$  pour toutes les fonctionnalités basées sur la Définition 1.5.3. Dans la Table 1.3, nous listons les résultats obtenus en appliquant notre approche sur la catégorie « Outils de communication » pour l'ensemble des plateformes e-learning considérées.
2. pour toutes les catégories de la Table 1.2, nous calculons les valeurs de  $prefMaxMin$  et  $prefMinMax$ . Les résultats des deux calculs sont affichés dans les Tables 1.4 et 1.5 respectivement.

TABLE 1.2 – Évaluation des fonctionnalités des plateformes d'e-Learning.

Catégorie	Outils de communication					Logiciel & Installation				Outils administratifs et Sécurité								Outils de présentation matériels				Caractéristiques de gestion			
	Chat	Forum	Mail	Vidéoconférence	Calendrier	Téléchargement	Installation	Assistance	Documentation	Cours d'administration	Suivi des progrès	Inscription en ligne	création de parcours pédagogique	Rapport	Organisation de parcours pédagogique	Test d'évaluation	Sécurité	Announces	Objects d'apprentissage	Exercices	Import de contenus	Gestion multi-cours	Gestion multi-utilisateurs	Gestion d'évaluation	Groupe d'utilisateurs
Critère	#	#	#	#	+	*	+	+	*	#	#	+	+	*	+	#	#	*	*	#	*	+	+	+	#
Claroline	*	+	+	*	*	#	+	+	*	+	E	+	+	E	+	#	#	*	*	#	*	+	+	+	+
Dokeos	—	#	#	+	+	*	#	#	#	—	*	#	#	*	#	#	#	—	*	#	*	+	+	E	—
eFront	#	+	+	+	+	+	—	—	—	—	+	#	—	—	—	—	#	0	+	+	+	+	+	+	—
ILIAS	#	+	+	0	+	+	—	—	—	—	+	—	—	—	—	#	#	0	#	+	0	+	+	+	—
Open ELMS	0	0	*	0	0	+	—	+	+	—	#	+	+	#	—	*	—	0	*	0	#	0	0	#	#
Ganesha	#	#	+	0	0	+	—	+	+	#	E	—	+	E	#	#	+	0	#	0	#	#	#	#	#
Olat	*	*	*	0	*	+	—	—	#	#	*	—	+	#	—	#	#	0	*	*	*	—	+	*	*
AnaXagora	#	#	#	0	+	+	—	+	—	#	*	—	*	+	#	#	+	0	+	#	—	—	#	#	*
Sakai	*	#	*	*	#	*	—	*	—	E	*	—	E	*	*	*	—	*	#	0	#	—	—	E	*

TABLE 1.3 – Résultats de *prefMin* et *prefMax* pour la catégorie « Outils de communication ».

Catégorie	Outils de communication					<i>prefMin</i>	<i>prefMax</i>
	chat	Forum	Mail	Vidéoconférence	Calendrier		
Critère							
<b>Claroline</b>	#	#	+	#	+	+	#
<b>Dokeos</b>	*	+	+	*	*	+	*
<b>eFront</b>		#	#	+	+		#
<b>ILIAS</b>	#	+	+	0	+	0	#
<b>Open ELMS</b>	0	0	*	0	0	0	*
<b>Ganesha</b>	#	#	+	0	0	0	#
<b>Olat</b>	*	*	*	0	*	0	*
<b>AnaXagora</b>	#	#	#	0	+	0	#
<b>Sakai</b>	*	#	*	*	#	#	*

Selon la Table 1.4, nous obtenons le classement suivant sur l'ensemble des systèmes d'e-learning considéré.

1. Claroline, Dokeos, eFront, Ganesha and Sakai.
2. Ilias, Open ELMS, Olat and AnaXagora.



TABLE 1.4 – Résultats du calcul de *prefMaxMin* sur l'ensemble des plateformes d'e-learning considérées.

Catégorie	Outils de communication	Logiciel & Installation	Outils administratifs et Sécurité	Outils de présentation matériels	Caractéristiques de gestion	<i>prefMaxMin</i>
Plateforme	<i>prefMin</i>	<i>prefMin</i>	<i>prefMin</i>	<i>prefMin</i>	<i>prefMin</i>	
<b>Claroline</b>	+	+	+	#	+	#
<b>Dokeos</b>	+			#	+	#
<b>eFront</b>		#				#
<b>ILIAS</b>	0			0		
<b>Open ELMS</b>	0			0	0	
<b>Ganesha</b>	0		+	0	#	#
<b>Olat</b>	0			0		
<b>AnaXagora</b>	0			0		
<b>Sakai</b>	#			0		#

Selon le tableau 1.5, nous obtenons le classement suivant sur l'ensemble des systèmes d'e-learning considéré.

1. Sakai
2. Claroline, Dokeos and Olat
3. eFront, ILIAS, Open ELMS, Ganesha and AnaXagora

TABLE 1.5 – Résultats du calcul de  $prefMinMax$  sur l'ensemble des plateformes d'e-learning considérées.

Categorie	Outils de communication	Logiciel & Installation	Outils administratifs et Sécurité	Outils de présentation matériels	Caractéristiques de gestion	$prefMinMax$
Plateformes	$prefMax$	$prefMax$	$prefMax$	$prefMax$	$prefMax$	$prefMinMax$
<b>Claroline</b>	#	*	*	*	#	#
<b>Dokeos</b>	*	#	<i>E</i>	*	<i>E</i>	#
<b>eFront</b>	#	*	*	+	+	+
<b>ILIAS</b>	#	+	#	#	+	+
<b>Open ELMS</b>	*	+	*	*	#	+
<b>Ganeshha</b>	#	+	<i>E</i>	#	#	+
<b>Olat</b>	*	#	*	*	*	#
<b>AnaXagora</b>	#	+	*	#	*	+
<b>Sakai</b>	*	*	<i>E</i>	*	<i>E</i>	*

Enfin, les utilisateurs peuvent faire un choix en fonction des opérateurs  $prefMaxMin$  ou  $prefMinMax$  ou peuvent combiner les résultats obtenus par les deux opérateurs. Par exemple, dans notre exemple illustratif, Claroline, Dokeos, eFront, Ganeshha et Sakai sont toutes des plateformes optimales selon l'opérateur  $prefMaxMin$ , tandis que Sakai est l'optimum selon l'opérateur  $prefMinMax$ . Mais, nous pouvons dire que Sakai la plus préférée par le groupe expérimental car elle est optimale selon les deux opérateurs.

## 1.6 Exemples de quelques systèmes tutoriels intelligents

Les STI permettent un entraînement interactif et en ligne des apprenants en faisant des exercices, des travaux pratiques (TP) ou des travaux dirigés (TD). Plusieurs domaines d'application ont adopté ces systèmes, surtout en apprentissage des langages de programmation comme Java, Python, PHP, HTML, etc. Dans cette section, nous avons décrit quelques STI dans le tableau 1.6. Par la suite, une comparaison de la situation d'apprentissage « Test » dans les plateformes de formation classique et dans les STI a été effectué dans le tableau 1.7 et dans le but de

prendre connaissance de fonctionnalités disponibles et de celles qui peuvent enrichir ce genre de systèmes. Nous avons étudié six systèmes de STI, à savoir :

- Online Python Tutor [Guo, 2013b];
- Coding Ground [Tutorials Point Team, 2016];
- Ideone [Sphere Research Labs, 2016];
- Regular Expression [Li *et al.*, 2008];
- CodeSkulptor [Warren *et al.*, 2014];
- Cloud9 [Ciortea *et al.*, 2010].

TABLE 1.6 – Description des différents STI

STI	Description	Type de licence	Langages
Online Python [Guo, 2013b]	Online Python c'est un outil éducatif gratuit créé par Philip Guo qui aide les utilisateurs à surmonter un obstacle donné et apprendre la programmation, la compréhension de ce qui se passe lorsque l'ordinateur exécute chaque ligne du code source d'un programme. Grâce à cet outil, nous pouvons écrire Python, Java, JavaScript, et les programmes Ruby dans le navigateur Web et de visualiser ce que l'ordinateur est en train de faire étape par étape.	Libre et open source	Python, Java, JavaScript, TypeScript, et Ruby
Coding Ground [Tutorials Point Team, 2016]	Coding Ground c'est un laboratoire en ligne où nous pouvons créer des programmes dans plus de 80 langages de programmation, de les compiler, les exécuter et de les partager sur le web. Il ne requiert qu'une connexion internet.	Libre et open source	80 langages
Ideone [Sphere Research Labs, 2016]	Ideone est un outil de compilation et de débogage en ligne qui permet aux apprenants de compiler le code source et de l'exécuter en ligne dans plus de 60 langages de programmation.	GNU GPL	plus de 60 langages
Regex testeur [Li <i>et al.</i> , 2008]	C'est un testeur d'expression régulière qui teste la façon avec laquelle un utilisateur d'ordinateur ou un programmeur s'exprime, comment un programme informatique devrait rechercher un modèle spécifique dans le texte, puis comment il fait chaque correspondance de modèle trouvé.	GNU General Public License version 2.0 (GPLv2)	Php, Javascript et Python
CodeSkulptor [Warren <i>et al.</i> , 2014]	C'est un interpréteur Python basé sur un navigateur web. Il met en œuvre un sous-ensemble de fonctionnalités de Python 2, ainsi que l'ajout de trois bibliothèques graphiques SimpleGui, SimpleMap et SimplePlot.	Open source	Python
Cloud9 [Ciortea <i>et al.</i> , 2010]	C'est un environnement de développement, dans le Cloud. Cloud9 combine un puissant éditeur de code en ligne avec un espace de travail Ubuntu complet dans le Cloud.	Gratuit	Plus de 40 langages

Contrairement aux plateformes d'apprentissage classique étudiées dans la section 1.5, les STI du tableau 1.6 contiennent les mêmes fonctionnalités suivantes :

- exécution du code ;
- édition du code ;
- chargement du code ;
- enregistrement du code.

Ces systèmes sont destinés à l'exécution d'un seul type de situation d'apprentissage qui est « travaux pratiques (TP) ». Les contenus d'apprentissage sont présentés sous forme d'exemples ; c'est à la charge des apprenants de créer des programmes selon leurs besoins et objectifs d'apprentissage comme ressources à exécuter. Nous avons remarqué aussi l'absence d'évaluation du travail de l'apprenant en termes de progrès et d'acquisition des connaissances. Ces systèmes vérifient la justesse du code de l'apprenant pendant l'exécution. Afin de recenser les avantages et les inconvénients d'apprentissage dans les STI ou les plateformes d'apprentissage classique, nous avons effectué une comparaison d'une situation d'apprentissage dans les deux types de systèmes dans la section suivante.

## **1.7 L'exercice dans les plateformes de formation classique et dans les STI**

Nous nous sommes basés sur l'évaluation des plateformes d'apprentissage classique faite dans la section 1.5 et sur le test de quelques systèmes STI dans la section 1.6 pour effectuer la comparaison de la situation d'apprentissage « Test » dans les deux types de systèmes. Nous avons pris en considération la situation « Test » dans la plateforme « Olat » [OLAT Team , Overall] qui a été jugée la meilleure plateforme au niveau de la classe hiérarchique « Hardware Présentation et outils », instance « Exercice », selon l'opérateur qui garantit la valeur maximale possible de satisfaction [Chachoua *et al.*, 2016b; Chachoua et Tamani, 2016]. Dans les STI considérés, la situation « Test » est une exécution d'un code dans un langage de programmation. Le tableau 1.7 montre la comparaison effectuée.

TABLE 1.7 – Comparaison entre la situation de « Test » dans les plateformes de formation classique et dans les STI

	plateforme de formation classique « Olat »			STI
Motif d'utilisation	Test examen	Auto-test exercice	Questionnaire sondage	un code à exécuter
Éditeur	éditeur de test	éditeur de test	éditeur de questionnaire	éditeur d'input/output
Types de questions	- choix unique - choix multiple - Kprim - texte à trous	- choix unique - choix multiple - Kprim - texte à trous	- choix unique - choix multiple - Kprim - texte à trous - texte libre	code en plusieurs langages de programmation
Intégration dans l'élément de cours	test	auto-test	questionnaire	indépendant du cours
Nombre de tentatives par participant	configurable	illimité	unique	illimité
Évaluation de l'apprenant	oui (statique)	oui (statique)	oui (statique)	non
Format & Norme du test	IMS : Instructional Management Systems QTI : Question and Test Interoperability (version 1.2)			
Conception du test & configuration d'évaluation	Formateur			C'est l'utilisateur qui exécute son code et le système compile et affiche le résultat.

D'après la comparaison de la situation « Test » dans les deux types de systèmes du tableau 1.7, nous avons tiré les conclusions suivantes :

### 1. Les plateformes d'apprentissage classique :

- l'existence de la notion de stratégie d'apprentissage ou la notion de scénarisation uniquement en « cours-test » car après chaque cours, un test d'évaluation est proposé sous forme de test, auto-test et/ou questionnaire. Ces types de test sont intégrés dans l'élément de cours selon les exigences des normes IMS et QTI ;
- le nombre de tentatives est configurable dans ces tests par un formateur ou le concepteur de la formation ;
- l'évaluation du travail de l'apprenant est faite d'une manière statique et configurable par le formateur.

### 2. Les STI :

- L'absence de la notion de stratégie d'apprentissage ou de la scénarisation de situations ;
- l'indisponibilité de ressources d'apprentissage structurées selon le niveau de difficulté de l'apprentissage. Il n'existe que des exemples illustratifs ;
- les tests effectués sont des programmes écrits par les apprenants qui sont exécutés dans un STI pour vérifier leur justesse. Ces tests sont indépendants du cours ;
- le nombre de tentatives dans ces tests est illimité, et il n'y a pas un paramétrage d'exécution, et ce, en raison de l'absence du formateur et/ou du concepteur du domaine ;
- aucune évaluation qualitative ou quantitative du travail de l'apprenant n'a été proposée.

Les inconvénients majeurs détectés dans l'apprentissage via ces deux types de systèmes sont les suivants :

1. l'inexistence de structuration en scénarii des situations d'apprentissage (Cours, TD, TP, Test) et leurs ressources associées ;
2. l'absence d'une évaluation dynamique et constructive du travail de l'apprenant qui prend en considération ses données dynamiques en termes de progrès et d'acquisition des connaissances ;
3. aucune adaptation de situations d'apprentissage ou de ressources n'a été proposée prenant en considération l'évaluation du travail de l'apprenant.

## 1.8 Discussion et conclusion

Dans ce chapitre 1, nous avons étudié les systèmes tutoriels intelligents en les définissant d'une manière générale, en détaillant leurs différentes entrées et sorties, leurs modèles (modèle de l'apprenant, modèle pédagogique, modèle de l'expert) qui constituent les composantes générales d'un STI ainsi que leurs catégories. Le choix de ces systèmes dans nos travaux est dû à l'efficacité de leur processus d'apprentissage. Ces systèmes d'apprentissage en ligne peuvent prendre en considération plusieurs caractéristiques et paramètres de l'apprenant comme son profil, ses préférences, ses besoins, son évolution et sa performance dans l'acquisition de connaissances pendant son cursus d'apprentissage. Nous avons détaillé les différentes catégories des STI pour se positionner dans les deux familles de la catégorie « modèle de traçage » et « modélisation basée sur les contraintes ». La première catégorie nous permet d'obtenir les traces de connaissances de l'apprenant durant l'apprentissage et leurs correspondances en termes de règles de production dans le modèle de domaine. La deuxième catégorie nous assure une modélisation de connaissances dans le domaine à base de contraintes en trois étapes : (a) une condition de pertinence qui indique quand la contrainte est applicable, (b) une condition de satisfaction qui teste l'état actuel de la solution de l'apprenant, et (c) un message de rétroaction lorsque l'état de la solution échoue à la condition de satisfaction.

Ensuite, dans le domaine d'évaluation des systèmes d'apprentissage en ligne en termes de performance pédagogique, plusieurs méthodes ont été proposées. Certaines se basent sur l'approvisionnement par la foule et qui peuvent rassembler des données formatives auprès d'une large base d'utilisateurs qui testent et vérifient un système d'apprentissage. Cependant, ce type souffre des problèmes d'adéquation des volontaires aux objectifs d'enseignement, la véracité de leurs réactions et de leurs réponses, et le sérieux avec lequel les volontaires interagissent avec le système. D'autres méthodes utilisent les apprenants simulés qui peuvent conduire à des prédictions d'apprentissage irréalistes. À moins de faire tester les systèmes d'apprentissage par des bénévoles représentatifs de la population cible partagés en deux groupes, un qui s'occupe des traitements et l'autre de la comparaison et du contrôle. Par ailleurs, l'analyse des trajectoires d'apprentissage donne le taux d'erreur dans les performances des apprenants par rapport à une estimation de leur niveau de connaissances et une modification de la trajectoire d'apprentissage en conséquence peut améliorer ou dégrader les performances de l'apprenant. D'autres méthodes comparent les décisions ou les choix d'un système intelligent automatisé à ceux d'experts humains. La plus récente méthode et qui peut être utile à nos travaux de recherches est l'exploration des données d'apprentissage des fichiers logs pour la modélisation et le profilage d'utilisateurs, la modélisation de domaine, et l'étude de composants d'apprentissage et de principes pédagogiques.

En revanche, il existe peu de méthodes d'évaluation des systèmes d'apprentissage en ligne en termes de fonctionnalités. Des méthodes comme QWS et AHP renvoient soit des valeurs



symboliques ou numériques ; ce qui peut être moins expressif et non intuitif du point de vue de l'utilisateur pour l'évaluation de la qualité d'un système. Nous avons proposé une approche hybride pour l'évaluation du système combinant les valeurs QWS, les relations de préférence symboliques et les opérateurs de comparaison formels, qui se sont révélés être des ordres totaux permettant de distinguer les systèmes d'apprentissage en ligne optimales du point de vue de l'utilisateur.

Nous avons étudié quelques STI afin de connaître les fonctionnalités disponibles dans ces systèmes. De plus, une comparaison entre une situation d'apprentissage « Test » dans les STI et les systèmes d'apprentissage classique a été effectuée dans le but de prendre connaissance des fonctionnalités disponibles et de celles qui peuvent enrichir ce genre de systèmes. Cette comparaison nous a révélé le manque d'une évaluation constructive basée sur les données dynamiques de l'apprenant qui peut nous permettre une adaptation de la situation ou de la ressource d'apprentissage.

Comme l'objectif de nos travaux de recherche est d'évaluer l'apprenant afin d'adapter le contenu d'apprentissage, nous étudions dans le chapitre suivant les méthodes développées pour l'évaluation de l'apprenant afin de connaître son niveau de connaissances et d'adapter les contenus pédagogiques en conséquence.

# Chapitre 2

## Évaluation de l'apprenant pour l'adaptation pédagogique

### 2.1 Introduction

Au fur et à mesure que la disponibilité des cours en ligne augmente, les concepteurs de cours et les instructeurs sont confrontés au défi de déterminer comment évaluer correctement l'activité d'apprentissage des apprenants [Palloff et Pratt, 2008]. L'évaluation fournit des informations précieuses concernant la motivation et les progrès des apprenants et peuvent être utilisées pour fournir des rétroactions pour les apprenants et les instructeurs. L'évaluation des connaissances et des compétences des apprenants du début à la fin d'un cours peut également déterminer l'efficacité d'un programme, et qui peuvent être la base des modifications aussi bien des contenus que des méthodes pédagogiques [Boston, 2002]. En général, les questions d'évaluation peuvent être ouvertes (réponses courtes ou essais) ou fermées (vrai/faux ou à choix multiples). Les questions ouvertes fournissent des informations approfondies sur la maîtrise des connaissances car elles exigent que les apprenants formulent des réponses, plutôt que d'identifier simplement les réponses correctes à partir d'une liste d'options possibles [Foddy, 1994].

L'évaluation des éléments ouverts repose généralement sur la connaissance et l'expertise des évaluateurs humains pour évaluer manuellement et faire des jugements personnels pour chaque réponse. Cette technique n'est pas toujours possible dans les programmes à grande échelle. Les interventions exigeants pour faire remédier à que les humains évaluent manuellement une grande quantité de réponses peuvent prendre un temps excessif et épuiser les ressources humaines [Williamson *et al.*, 2012; Landauer *et al.*, 2003]. En outre, la mise en place de telles demandes est potentiellement lourdes sur les évaluateurs humains et peut augmenter la probabilité d'erreur [Ramineni et Williamson, 2013]. En réponse aux limitations pratiques associées à l'évaluation manuelle, les techniques d'évaluation automatisée peuvent offrir une solution possible.

Nous étudions dans ce chapitre les méthodes d'évaluation des apprenants et les systèmes d'adaptation des ressources et stratégies d'apprentissage. La section 2.2 détaille les méthodes d'évaluation automatisée des apprenants, leurs objectifs et les paramètres nécessaires pour la mise en œuvre. Par ailleurs, nous détaillons les techniques d'évaluation automatisées qui sont utilisées dans les systèmes d'évaluation automatisés comme le Service de Test Éducatif (STE) qui utilise l'Évaluation Automatisée des Essais (EAS). Par la suite, dans la section 2.3 nous nous intéressons à ces systèmes dans notre domaine d'étude : les systèmes tutoriels intelligents (STI). Notre principal objectif est d'étudier l'adaptation à base de ces systèmes d'évaluation. Dans ce contexte, la section 2.4 détaille les évaluations adaptées qui ont été proposées dans plusieurs domaines d'application. Dans la section 2.5, nous nous intéressons à l'adaptation de la ressource et/ou du trajectoire d'apprentissage dans les systèmes d'apprentissages. Parmi ces derniers, dans la section 2.6 nous nous concentrons sur l'adaptation dans les STI. À partir de l'état de l'art effectué, nous avons établi dans la section 2.7 un bilan qui englobe les limitations des méthodes d'évaluation et celles des systèmes d'adaptation. Dans la section 2.8, nous clôturons le chapitre avec une conclusion.

## 2.2 Méthodes d'évaluation automatisées des apprenants

L'évaluation automatique est considérée comme un outil important dans les systèmes éducatifs comme détaillé dans le rapport de la Communauté de l'Union Européenne [Scheuermann et Björnsson, 2009]. Ce dernier est un recueil d'articles et d'études sur le sujet de l'évaluation de divers points de vue englobant les besoins d'évaluation dans les pays européens.

Dans [Crisp, 2009], trois types d'évaluation et leurs relations ont été introduites, à savoir :

- diagnostic : exécuté avant l'apprentissage pour estimer le niveau des élèves ;
- évaluation formative : appliquée lors de l'apprentissage pour tester l'évolution des connaissances ;
- évaluation sommative : considérée après l'apprentissage pour évaluer les connaissances acquises.

Ces types d'évaluation permettent aux apprenants de suivre leurs progrès ainsi que leurs objectifs ciblés au cours d'une activité donnée, afin de savoir s'ils ont réussi ou échoué un niveau de formation donné. Dans [Amelung *et al.*, 2011], les auteurs ont conçu et mis en place un support générique d'évaluation pour les devoirs dans les langages de programmation qui pourrait être intégré dans les STI [Guo, 2013a]. Ils ont également développé un module appelé « edu-Components » pour l'évaluation des apprenants [Amelung *et al.*, 2006, 2007]. Dans [Nicol et Macfarlane-Dick, 2006; Nicol, 2007], une vaste étude d'évaluation formative et de rétroaction (feedback) a été réalisée dans le but d'identifier la façon dont les processus d'évaluation

peuvent aider à la mise en valeur et le développement de l'auto-direction et la réglementation de l'apprenant. La norme SCORM [Scorm, 2006] considère à la fois la durée et les tentatives dans les activités de test telles que celles mises en œuvre dans des plateformes d'e-learning, cours de certification et les examens d'IT, les essais de recrutement, etc.

Au mieux de nos connaissances, les fonctions d'évaluation n'ont été considérées que très peu. Par exemple, les auteurs dans [Burstein *et al.*, 2001] ont défini la fonction d'évaluation dans un contexte d'essai comme une fonction linéaire et dans [Yang *et al.*, 2002], les auteurs considèrent que les scores automatisés sont cohérents avec les scores des experts évaluateurs humains. Dans [Kozma, 2009] l'auteur indique qu'il est nécessaire de proposer de nouvelles théories et de nouveaux modèles de notation des processus et des stratégies des apprenants au cours des évaluations. Cependant, le domaine d'étude de l'évaluation automatisée reste assez nouveau et les lignes directrices des meilleures pratiques évoluent encore [Williamson *et al.*, 2012]. Les essais et les résultats de tests selon Wandall [Wandall, 2011] peuvent être utilisés afin d'évaluer les connaissances et les compétences de l'apprenant, l'effet de l'enseignement mais aussi pour l'admission, la réglementation, le contrôle, la gratification ou punition des individus dans les écoles.

L'évaluation précoce des apprenants est importante dans tout processus pédagogique. Son rôle est double. D'abord, elle donne la possibilité de quantifier le niveau de l'apprenant et donc d'adapter de façon aussi optimale que possible les ressources d'apprentissage et procéder en conséquence. Ensuite, elle peut identifier les modèles de progrès de l'apprenant après certaines sessions de formation. Par conséquent, une évaluation de l'apprenant par un système sophistiqué qui exploite des données supplémentaires appelées traces a été proposée dans [Kumar *et al.*, 2015; Papamitsiou et Economides, 2015]. Dans [Papamitsiou et Economides, 2015], une visualisation des traces temporelles au cours de l'évaluation a été proposée pour aider les acteurs impliqués dans l'activité à interpréter intuitivement et percevoir rapidement une fonctionnalité cachée. Dans [Kumar *et al.*, 2015], une plateforme en ligne appelée « Learnform » a été développée pour « Problem Solving » dans le domaine de l'électricité et magnétisme. L'évaluation du progrès de chaque apprenant est basée sur leurs traces, sur lesquelles l'approche statistique de « Cohen » a été appliquée pour comparer entre les scores de post-test de l'apprenant avec son pré-test et de son état, pour vérifier s'il a amélioré ou non ses connaissances [Cohen, 1977].

Les systèmes automatisés d'évaluation, dans lesquels les réponses sont notées par des machines, ont été utilisés pour évaluer les réponses courtes, les essais et les réponses orales. Par exemple, le Service de Test Éducatif (STE) a utilisé l'Évaluation Automatisée des Essais (EAS) pour des notations, comme le GMAT (Graduate Management Admission) ou le GRE (Graduate Record Examination) [Williamson *et al.*, 2012]. Différents systèmes EAS existent, qui requièrent un grand nombre d'essais pour fonder leurs notations et leurs rétroactions. Ces systèmes ont tendance à fournir des rétroactions à la fois holistiques et spécifiques, bien que le

contenu précis varie selon le système. Dans l'ensemble, les systèmes EAS ont été jugés valides et fiables [Dikli, 2006].

L'application de techniques automatisées d'évaluation pour les réponses courtes est moins contestées, surtout lorsque ces techniques ne sont pas utilisées pour évaluer globalement la qualité de l'écriture avec des implications importantes. Les systèmes automatisés pour les réponses courtes sont les mieux adaptés pour mesurer des concepts, des faits ou des compétences explicites [Brew et Leacock, 2013]. Le système automatisé le plus connu pour les réponses courtes est « C-rater » qui a été développé par ETS (Educational Testing Service) [Leacock et Chodorow, 2003]. La validité de C-rater a été évaluée en comparant les scores automatisés avec les scores de l'humain. Il a été constaté que les scores automatisés correspondaient aux scores attribués par les évaluateurs humains à 84% [Leacock et Chodorow, 2003]. Des résultats similaires ont été reproduits ailleurs, ce qui suggère que les systèmes automatiques pour les réponses courtes peuvent offrir un moyen d'évaluation valable [Burstein *et al.*, 2003].

Malgré le fait que la technologie pour la création et l'utilisation de systèmes automatisés existe depuis au moins une décennie, un nombre limité de ce type de système a été développé et diffusé à travers les domaines.

S'appuyant sur ces travaux antérieurs, nos recherches ont porté sur l'évaluation dans les STI.

## 2.3 Évaluation dans les Systèmes Tutoriels Intelligents

Les systèmes de tutorat intelligents (STI) prennent de plus en plus d'importance dans l'éducation. Beaucoup de recherches ont été menées pour représenter et modéliser les connaissances des apprenants avec précision, concevoir des programmes d'études efficaces et élaborer des stratégies d'enseignement optimales. Un gros travail s'est concentré sur l'extraction des audits de données (logs) collectés par les STI. Un sujet important dans ce domaine est l'évaluation automatique de l'apprentissage de l'apprenant et la détection des propriétés des apprenants, par exemple les aptitudes mentales, les troubles d'apprentissage, etc. [Shute, 2011]. Les évaluations traditionnelles prennent souvent du temps et doivent être supervisées par un expert, ce qui les rend coûteuses en pratique. Par conséquent, l'évaluation traditionnelle ne passe pas l'échelle et n'est donc pas appropriée dans de nombreux cas, comme dans les MOOC, les grands cours universitaires ou les projections répandues dans les écoles élémentaires pour permettre la détection précoce des troubles d'apprentissage.

Des travaux antérieurs ont étudié des évaluations numériques automatiques autonomes, y compris l'évaluation automatique [Attali, 2015], la génération de contenus pédagogiques [Graf et Fife, 2012] et l'évaluation par jeu [Hao *et al.*, 2015]. En outre, des programmes de dépistage numérique remplaçant les tests neuropsychologiques traditionnels, par exemple pour la dyscalculie [Butterworth, 2003] ou la dyslexie [Cisero *et al.*, 1997], ont été développés. Ces pro-

grammes de dépistage par ordinateur sont parfaitement intégrables dans un STI. Cela permet non seulement l'évaluation automatique et non intrusive des apprenants, mais aussi l'analyse et la détection des traits des apprenants qui permettent une meilleure adaptation du curriculum aux besoins individuels. Une étape dans cette direction est la détection de comportements intégrés qui identifient les apprenants qui jouent dans le système [Baker *et al.*, 2004], trouvant des apprenants qui tournent en rond [Beck et Gong, 2013] ou la modélisation de l'implication effective des apprenants [Arroyo et Woolf, 2005; von Aster *et al.*, 2006; Joseph, 2005]. D'autres travaux ont utilisé des approches de regroupement et de classification pour détecter les caractéristiques mathématiques des apprenants [Käser *et al.*, 2013b]. Un autre travail propose un pipeline pour intégrer l'évaluation automatique, c'est-à-dire les détecteurs de traits d'apprenants, directement dans le système de tutorat [Klingler *et al.*, 2016]. L'approche a été validée dans le cas du développement de la dyscalculie pour détecter le trouble d'apprentissage spécifique à l'acquisition de compétences arithmétiques [Von Aster et Shalev, 2007] et l'environnement de formation basé sur le jeu « *Calcularis* » [Käser *et al.*, 2013a].

Un mécanisme automatique d'évaluation a été développé dans un système de tutorat intelligent pour l'enseignement de l'intelligence artificielle (SEIA) [Grivokostopoulou et Hatzilygeroudis, 2013]. Ce mécanisme interagit avec le mécanisme de détection d'erreur pour savoir si la réponse est correcte ou non. Deux types d'exercices-réponses sont considérés dans l'évaluation : simple et complexe. Les réponses sont présentées sous forme de nœuds d'états dans un arbre. Les exercices dont les réponses correctes sont inférieures ou égales à 6 nœuds-états sont considérés comme des exercices simples. Le reste des exercices est considéré comme complexe. Initialement, le mécanisme calcule la distance d'édition entre la séquence d'une simple réponse de l'apprenant et celle de la réponse correcte d'un expert [Grivokostopoulou *et al.*, 2017].

Nous constatons que des mécanismes d'évaluation automatique ou numérique ont été développés pour remédier aux limitations des systèmes d'évaluations traditionnels. Ces méthodes d'évaluations ont été considérées dans d'autres situations d'apprentissage telle que la détection de troubles d'apprentissage afin d'éviter l'abandon dans un processus de formation. Ces méthodes sont dites d'évaluation adaptées et sont détaillées dans la section suivante.

## 2.4 Évaluations adaptées

L'objectif de ces méthodes d'évaluation est d'adapter les questions posées à un apprenant en se basant sur son évaluation durant une activité d'apprentissage. Au lieu de donner les mêmes questions à tous, les tests informatiques adaptatifs ((TIA) CATs-Computer Adaptive Tests) sélectionnent la prochaine question à poser en fonction des réponses précédentes, permettant ainsi une adaptabilité à chaque étape [van der Linden et Glas, 2010]. La conception des TIA repose

sur deux critères : un premier critère de terminaison, par exemple, un certain nombre de questions à poser et un deuxième critère du prochain élément. Bien que le critère de terminaison ne soit pas satisfait, les questions sont posées selon le critère du prochain élément, qui choisit des questions, par exemple, les plus informatives sur la capacité ou la connaissance de l'apprenant. De tels tests adaptatifs nécessitent moins de questions que les tests non adaptatifs pour atteindre la même précision de prédiction en termes d'objectifs [Lan *et al.*, 2014].

Compte tenu d'un modèle d'apprenant [Peña-Ayala, 2014], l'objectif est de fournir une mesure précise des paramètres d'un apprenant tout en minimisant le nombre de questions posées. Ce problème a été appelé réduction de taille du test [Lan *et al.*, 2014], et il est également lié à l'évaluation du rendement des apprenants [Bergner *et al.*, 2012; Thai-Nghe *et al.*, 2011]. Dans l'apprentissage par machine, cette approche est connue sous le nom d'apprentissage actif qui permet d'interroger de manière adaptée les étiquettes informatives d'un ensemble de formation afin d'optimiser l'apprentissage. Un autre facteur important consiste en la longueur des tests en ce sens que les tests plus courts sont meilleurs pour le système, car ils réduisent la charge, et sont meilleurs pour les apprenants, qui peuvent être frustrés ou ennuyés s'ils doivent donner trop de réponses [Chen *et al.*, 2015; Lynch et Howlin, 2014]. Ainsi, l'évaluation adaptative est de plus en plus utile à l'état actuel des MOOC, où la motivation joue un rôle important [Lynch et Howlin, 2014]. Cependant, dans les scénarii de la vie réelle il faut tenir compte de plus de contraintes. Tout d'abord, le calcul des critères doit être effectué dans un délai raisonnable d'où la complexité importante de ces approches. Deuxièmement, les compétences d'évaluation doivent être effectuées sous l'incertitude : un apprenant peut patiner, c'est-à-dire, accidentellement ou négligemment, échouer dans un élément qu'il aurait pu résoudre, ou deviner, c'est-à-dire répondre correctement à un élément par hasard. C'est pourquoi l'évaluation adaptative ne peut pas simplement effectuer une recherche binaire sur la capacité de l'apprenant en posant une question plus difficile s'il réussit et une question plus facile s'il échoue. Ainsi, nous devons utiliser des méthodes plus robustes, comme les modèles probabilistes pour l'évaluation des compétences.

Plusieurs modèles peuvent être utilisés, selon le but de l'évaluation adaptative, par exemple, l'estimation d'un niveau général de compétences, la fourniture d'informations de diagnostic ou la caractérisation des connaissances [Mislevy *et al.*, 2012]. À la fin du test, une rétroaction peut aider les enseignants à identifier les apprenants en difficulté. Il protège également contre les erreurs de persévérance lorsque les apprenants répondent de manière incorrecte à un test de pratique [Dunlosky *et al.*, 2013]. Par exemple, un test pratique à choix multiples augmente les intrusions de fausses alternatives lors d'un test de rappel final et lorsque aucune rétroaction n'est fournie, alors qu'aucune amélioration n'est prévue durant l'apprentissage [Butler *et al.*, 2008]. Toutefois, l'effet correctif de la rétroaction n'est pas obligatoire immédiatement après le test de pratique. Les auteurs dans [Metcalfe *et al.*, 2009] précisent que la performance du test

final pour les réponses initialement incorrectes était en fait meilleure lorsque les commentaires avaient été retardés que lorsqu'ils avaient été immédiats.

Dans nos travaux on s'intéresse aux modèles d'adaptation basés sur l'évaluation des apprenants durant une activité d'apprentissage. Dans ce qui suit, nous présentons un état de l'art des systèmes d'adaptation dans les plateformes d'apprentissage en ligne.

## 2.5 Systèmes d'adaptation dans les plateformes d'apprentissage en ligne

Les systèmes adaptatifs sont d'une grande aide pour les apprenants et les enseignants dans les systèmes d'apprentissage en ligne. Ces systèmes peuvent être utilisés dans différents contextes, tels que dans les systèmes pédagogiques adaptatifs. L'adaptation est une approche de personnalisation basée sur le sujet développé pour adapter l'annotation du contenu en fonction du profil de l'apprenant et l'acquisition de connaissances pour améliorer l'enseignement comme par exemple des langages de programmation [Brusilovsky *et al.*, 1999, 1998; Weber et Specht, 1997].

Dans un environnement d'e-learning basé sur le web, une plateforme adaptative appelée « Lecomps » a été développée [Sterbini et Temperini, 2009]. Cette dernière automatise le processus de construction des trajectoires adaptées d'apprentissage personnalisés, en tenant compte des objectifs d'apprentissage, des connaissances de l'apprenant et des styles d'apprentissage individuels. Il s'appuie sur un moteur basé sur la logique de contraintes pour construire des objets d'apprentissage (LOs-Learning Objects) tels que définis dans [Felder et Silverman, 1988; Felder et Spurlin, 2005]. La description de LOs correspond aux Taxonomies de Bloom et d'ACM [Bloom's, 1965; Gorgone *et al.*, 2002]. Le processus d'adaptation est basé sur le niveau de l'apprenant lié aux connaissances acquises et requises pour passer d'un niveau à un autre.

L'adaptation est utilisée pour générer des parcours d'apprentissage personnalisés basés sur le statut d'apprentissage des apprenants [Hwang *et al.*, 2010]. Une approche de génération de trajectoires d'apprentissage adaptatives a été initialement introduite dans les Systèmes de Gestion de l'Apprentissage (SGA) et LMS, plus exactement dans la plateforme Moodle (2007) [Graf, 2007], basée sur le calcul d'un score d'apprentissage a été proposée dans [Li *et al.*, 2003]. L'approche considère un nombre total de niveaux ( $L$ ) et de difficultés de niveaux ( $d$ ) pour chaque activité d'apprentissage.

Dans [Göbel *et al.*, 2009], les auteurs introduisent un modèle d'adaptation dans le domaine du Jeu Éducatif (JE). Deux modèles d'adaptation sont définis, à savoir : (i) « macro-adaptation » qui est une adaptation de la séquence de scène, et (ii) « micro-adaptation » qui est une adaptation d'éléments de scène, qui sont utilisés dans [Gros *et al.*, 2015; Zarraonandía *et al.*, 2016]



pour développer un modèle d'adaptation étendu en définissant un ensemble de règles et de scénarii à un jeu et aux apprenants. Ce modèle d'adaptation permet de modifier une séquence prédéfinie et de modifier le niveau de difficulté en activant/désactivant des règles de jeu. Par exemple, la macro-adaptation est implémentée dans le jeu éducatif « Elektra » qui est un jeu d'aventure en 3D qui permet d'enseigner aux apprenants la physique de l'optique [Peirce *et al.*, 2008]. Pendant le jeu, le joueur doit surmonter différents défis liés à ce sujet, qui peuvent prendre différentes formes, telles que résoudre un puzzle, effectuer une tâche ou manipuler un appareil. Tous les défis sont présentés et intégrés dans le cadre d'une histoire significative. Le système utilise la base de règles et les méthodes probabilistes pour interpréter les événements dans le jeu et les transformer en preuves sur l'apprenant.

Il est à noter que la norme SCORM considère à la fois la durée et les tentatives dans les activités de test comme paramètres à établir pour un test ou un examen, afin de mettre les apprenants sous un stress temporel en imposant des réponses à des questions ou résoudre des puzzles dans un temps donné [Scorm, 2006]. La norme SCORM a été étendue dans [Rey-López *et al.*, 2009] pour proposer ADL-SCORM permettant une création de cours adaptatifs basés sur le profil utilisateur. L'adaptation a été prise en compte à deux niveaux, à savoir, les objets de contenu utilisables (SCO-Sharable Content Objects) et l'activité. Les deux adaptations dépendent des paramètres déduits du profil de l'utilisateur. Basé sur la norme SCORM, la plateforme « T-Maestro » offre un ensemble de services pour offrir de l'apprentissage personnalisé à l'apprenant (dans ce cas, le spectateur), en utilisant la télévision comme support de diffusion. L'adaptation se fait à deux niveaux différents : l'adaptation au niveau SCO permet aux SCOs de modifier leur comportement en fonction des paramètres d'adaptation déduits du profil de l'apprenant, alors que l'adaptation au niveau de l'activité permet d'établir différentes manières d'atteindre les objectifs d'une activité en fonction des valeurs des paramètres d'adaptation pour l'apprenant qui étudie le cours.

IMS Learning Design (IMS LD) par opposition à IMS (Simple Sequencing), SCORM se focalise sur l'organisation des activités et non pas sur l'organisation du système. IMS-LD est un langage de modélisation pédagogique. Dans [Burgos *et al.*, 2007], l'auteur examine huit types d'adaptation, à savoir : adaptation basée sur l'interface, adaptation basée sur l'apprentissage, adaptation basée sur le contenu, support interactif de résolution de problèmes, le filtrage adaptatif d'informations, le regroupement adaptatif des utilisateurs, l'évaluation adaptative et les changements à la volée.

Le premier effort de la TLA, l'expérience de l'API , ou xAPI [Adv, 2014] permet la traçabilité dans les plateformes.

La personnalisation de l'adaptation des contenus didactiques et des parcours de navigation à travers les ressources d'apprentissage a été considérée comme une question importante pour le développement des systèmes d'apprentissage efficaces en ligne [Van Seters *et al.*, 2012]. Nous pouvons distinguer deux types d'adaptation, à savoir : (i) « présentation adaptative » qui génère

un contenu personnalisé pour un apprenant individuel, en s'appuyant sur les styles d'apprentissage des apprenants, et (ii) « navigation adaptative » qui guide les apprenants à travers des trajectoires d'apprentissage personnalisées pour naviguer dans un ensemble de contenus d'apprentissage appropriés [Brusilovsky, 2001]. Le processus d'adaptation tient compte de différents paramètres tels que les caractéristiques ou préférences des apprenants et/ou des contenus d'apprentissage [Wang et Wu, 2011], souvent considérés comme des ressources. Peu de travaux ont utilisé les traces pour l'adaptation. Nous pouvons citer [Zouhair *et al.*, 2013] dans le domaine de l'Environnement Informatique pour l'Apprentissage Humain (EIAH), dans lequel un système de suivi des apprenants a été conçu et mis en œuvre pour assurer un suivi individualisé et continu de l'apprenant pendant le processus d'apprentissage. Afin de détecter les causes des difficultés et abandon des apprenants et agir en conséquence, il est nécessaire de s'appuyer sur un raisonnement basé sur la trace de chaque situation d'une manière dynamique. Le processus d'individualisation est mis en œuvre dans un système multi-agent dans lequel les agents aident ceux en difficulté en leurs proposant une situation correcte ou adaptée. Plusieurs processus d'adaptation ont été proposés dans plusieurs domaines d'application et dans différents contextes pour implémenter des systèmes adaptatifs d'apprentissage tels que [Mustafa et Sharif, 2011; Sosnovsky et Brusilovsky, 2015; Sterbini et Temperini, 2009; Hwang *et al.*, 2010]. Des systèmes hypermédia d'e-learning adaptatifs ont été proposés [Mustafa et Sharif, 2011] qui exploitent les deux types d'adaptation susmentionnés pour mettre en œuvre les styles d'apprentissage et leurs dimensions [Felder et Silverman, 1988; Felder et Spurlin, 2005].

Le Tableau 2.1 résume les systèmes d'adaptation dans les plateformes d'apprentissage en ligne et les normes qui ont contribué à l'adaptation de la ressource et/ou la stratégie d'apprentissage aux apprenants.

TABLE 2.1 – Tableau synthétique sur les méthodes d'adaptation existantes

<b>Plateforme d'apprentissage en ligne</b>	<b>Adaptation</b>	<b>Méthodes d'adaptation</b>	<b>Évaluation (Score)</b>	<b>Traces</b>
Apprentissage du langage de programmation [Sosnovsky et Brusilovsky, 2015]	Adapter l'annotation du contenu	Adaptation basée sur le profil de l'apprenant et son acquisition	Non	Non

Plateforme d'apprentissage en ligne	Adaptation	Méthodes d'adaptation	Évaluation (Score)	Traces
Les systèmes d'e-learning [Brusilovsky <i>et al.</i> , 1999, 1998; Weber et Specht, 1997; Van Seters <i>et al.</i> , 2012; Brusilovsky, 2001; Mustafa et Sharif, 2011; Sterbini et Temperini, 2009]	Adapter et personnaliser les chemins d'apprentissage	- Adaptation basée sur les objectifs pédagogiques, - les connaissances de l'apprenant et ses styles d'apprentissage ; - Utilisation de la logique de contrainte pour construire les LOs.	Non	Non
Monde réel [Hwang <i>et al.</i> , 2010]	Adapter des parcours d'apprentissage personnalisés	Adaptation basée sur le statut d'apprentissage des apprenants	Non	Non
Systèmes de Gestion de l'Apprentissage (SGA) [Graf, 2007]	Navigation adaptative	Adaptation basée sur le calcul d'un score d'apprentissage	Oui mais une évaluation statique	Non
Jeu Éducatif (JE) [Göbel <i>et al.</i> , 2009; Gros <i>et al.</i> , 2015]	Macro-adaptation et micro-adaptation	Adaptation basée sur l'activation/désactivation des règles de jeu pour modifier le niveau de difficulté	Non	Non
La norme ADL-SCORM [Rey-López <i>et al.</i> , 2009] et IMS-LD [Burgos <i>et al.</i> , 2007]	- Adaptation des objets de contenu utilisables (SCO) et des activités d'apprentissage, - Adaptation du séquençage et navigation des activités d'apprentissage.	- Adaptation basée sur le profil utilisateur, - Adaptation centrée sur la notation des objectifs.	Oui mais la méthode de notation n'est pas définie	Non

Plateforme d'apprentissage en ligne	Adaptation	Méthodes d'adaptation	Évaluation (Score)	Traces
Environnement Informatique pour l'Apprentissage Humain (EIAH) [Zouhair <i>et al.</i> , 2013]	- Adaptation des situations d'apprentissage	-Adaptation basée sur la trace de chaque situation	Non	Oui

## 2.6 Adaptation dans les Systèmes Tutoriels Intelligents

L'apprentissage peut être défini comme des processus internes de changement, en raison de l'expérience personnelle de l'apprenant. Aussi, il peut être défini comme l'acquisition ou l'ajout de quelque chose de nouveau, ce qui implique une variation ou modification préalablement acquise [Rivas Navarro, 2008]. Les enseignants guident les apprenants pendant l'apprentissage et cherche à savoir si ces derniers améliorent leurs niveaux de connaissances. Cependant, dans les environnements de tutorat d'une classe classique, le temps dédié par un formateur à chaque apprenant individuellement diminue considérablement. Pour cette raison, certains travaux proposent l'utilisation d'un système logiciel pour satisfaire ces besoins [Carbonell, 1970; Cataldi et Lage, 2010; Huertas et Juárez-Ramírez, 2013; Santhi *et al.*, 2013]. Cependant, le système logiciel doit être adapté aux besoins de l'apprenant.

L'adaptabilité aux besoins des étudiants est un défi pour l'ingénierie logicielle [Luckey et Engels, 2013]. L'adaptabilité est définie comme l'adaptation du logiciel aux caractéristiques individuelles des utilisateurs selon leurs objectifs [Radenković *et al.*, 2011; Boukhari *et al.*, 2018]. Différents types d'adaptations logicielles sont définis, mais ces recherches portent sur l'adaptation du contenu [Radenković *et al.*, 2011; Razek et Bardesi, 2013; Barac *et al.*, 2015]. Le logiciel a besoin des données de l'utilisateur et ses caractéristiques pour qu'il produit une adaptation de contenu. Cet ajustement de l'environnement d'apprentissage peut être réalisé avec des stratégies intelligentes [Razek et Bardesi, 2013] pour déduire les besoins des utilisateurs. Un type particulier de logiciel répondant aux caractéristiques mentionnées ci-dessus s'appelle Système de Tutorat Intelligent (STI), qui peut être défini comme un système logiciel qui utilise des techniques intelligente pour interagir avec les apprenants à enseigner de la même manière qu'un enseignant [Cataldi et Lage, 2010; Huertas et Juárez-Ramírez, 2013; Santhi *et al.*, 2013].

Un problème important dans le développement des (STI) réside dans l'évaluation des connaissances des apprenants [Conejo *et al.*, 2001]. Les STI doivent pouvoir déterminer avec précision et rapidité le niveau cognitif de l'apprenant pour décider ce qui est important pour les

enseigner. Pour cela, la théorie des probabilités a été proposée par certains auteurs pour gérer l'incertitude dans le diagnostic des connaissances des apprenants [Conejo *et al.*, 2001; Huertas et Juárez-Ramírez, 2013; Santhi *et al.*, 2013]. Un travail basé sur les réseaux bayésiens propose une architecture générale d'un STI pour implémenter un module d'évaluation des connaissances [Ramírez-Noriega *et al.*, 2017]. L'objectif est de diagnostiquer efficacement les besoins d'apprentissage des apprenants et renforcer les sujets faibles. Le modèle probabiliste Bayésien est utilisé pour la gestion de l'incertitude et la représentation du savoir. Il est basé sur la même compréhension fondamentale du domaine et capable de résoudre différentes tâches, telles que l'identification, l'évaluation, le suivi et le risque d'atténuation [Liu et Wang, 2007; Misirli et Bener, 2014; Santhi *et al.*, 2013]. Les réseaux bayésiens utilisent une inférence flexible basée sur des observations partielles, ce qui permet de raisonner sur les données des utilisateurs pendant leurs interactions avec le système. Cependant, les modèles de réseaux bayésiens nécessitent de nombreux paramètres numériques. S'il n'y a pas de données disponibles ou si les données disponibles sont incomplètes, cela peut entraîner des problèmes de paramétrage du modèle. Le tableau 2.2 détaille les Systèmes d'adaptation dans les STI.

TABLE 2.2 – Tableau synthétique sur quelques méthodes d'adaptation dans les STI

Système STI	Adaptation	Méthode d'adaptation	Traces
Tuteurs cognitifs [Santhi <i>et al.</i> , 2013; Ramírez-Noriega <i>et al.</i> , 2017; Liu et Wang, 2007; Misirli et Bener, 2014; Santhi <i>et al.</i> , 2013]	- Modèle cognitif de l'apprenant, - Modèle du domaine et résolution de problèmes.	Réseaux bayésiens	Oui
Adapt Case Modeling Language (ACML) [Luckey et Engels, 2013]	- Auto-optimisation, - Auto-protection, - Auto-configuration, - Auto-guérison.	Décrite par les self-propriétés	Oui
	- Adaptation de la structure du système, - Adaptation du comportement.	- Adaptation du type (changement permanent), - Adaptation des instances (changement temporaire) d'un modèle.	
	- Adaptation réactive, - Adaptation proactive est partiellement soutenue par l'utilisation d'historiques spécifiques.	Mécanismes de modélisation.	

Système STI	Adaptation	Méthode d'adaptation	Traces
Web portal for adaptive e-learning- Moodle [Radenković <i>et al.</i> , 2011]	- Connaissance du domaine, - Styles d'apprentissage.	Mécanismes d'adaptation sous forme de services : - Service de définition des objectifs et des contraintes, - Service de création, visualisation et gestion de questionnaire, - Service de regroupement des apprenants en groupes. - Service de présentation des objets d'apprentissage adaptés.	Oui
Adaptive Course for Mobile Learning/Adaptive Social Mobile Learning System (ASMLS) [Razek et Bardesi, 2013]	- Objets d'apprentissage adaptatifs.	Exploite deux types d'agent : - Agent de filtrage vise à obtenir différents types de connaissances des apprenants, - Agent pédagogique utilise ces connaissances pour l'adaptation.	Non

## 2.7 Bilan

À travers l'état de l'art sur les modèles d'évaluation des apprenants et les méthodes d'adaptation de ressources et de stratégies d'apprentissage, nous constatons qu'il existe beaucoup de méthodes d'évaluation des apprenants dans différents domaines d'application. La multitude de ces méthodes a contribué à la diversification des méthodes d'adaptation selon le but voulu de l'évaluation. En dépit de leurs avantages manifestes, il y a quelques limitations fondamentales.

Dans les systèmes « Computing Adaptive Testing (CAT) » et les méthodes de « computer-based testing » et « benchmarking », les auteurs de l'étude [Kozma, 2009] ont souligné la limitation des méthodes d'évaluation traditionnelles comme outils de mesure des compétences et des capacités maîtrisées par les apprenants. En ce qui concerne l'évaluation des devoirs dans le domaine des langages de programmation, le module proposé ne fournit qu'une évaluation semi-automatique et statique, réalisée par le tuteur sous la direction de l'outil [Guo, 2013a]. Ainsi, les évaluations formatives et sommatives sont effectuées par l'instructeur. Notre approche propo-

sée peut être considérée comme une amélioration de ces méthodes et comme une étape vers un système d'évaluation adaptatif (computer adaptive testing) basé sur les traces des apprenants.

Par ailleurs, dans l'apprentissage en ligne, les questions ouvertes peuvent fournir des informations plus utiles, mais malheureusement elles sont plus difficiles à évaluer [Landauer *et al.*, 2003]. Cependant dans [Brew et Leacock, 2013], l'auteur a suggéré que la notation automatisée des réponses courtes est plus difficile à créer que l'Évaluation Automatisée des Essais (EAS), car il tend à se concentrer sur la grammaire et la technique d'évaluation, tandis que la notation automatisée de la réponse a tendance à se concentrer sur le contenu. Les systèmes automatisés de notation de réponse courte sont sous-utilisés car il est actuellement impossible de concevoir un moteur de notation de réponse courte, qui fonctionnera pour tous les éléments de réponses [Brew et Leacock, 2013]. En particulier, l'EAS a été critiqué pour sa simplification de l'évaluation de l'écriture et sa concentration sur des éléments de routine, tels que le nombre de mots ou la complexité du choix des mots, au lieu des aspects difficilement quantifiables, tels que la réflexion de la réponse ou l'écriture pour un public spécifique [Condon, 2013; Perelman, 2014].

La majorité des systèmes d'apprentissage en ligne est fondée sur des normes comme SCORM, IMS-LD, AICC, etc. Bien que la norme SCORM inclue un modèle de séquençage et de navigation centrée sur la notation des objectifs, il ne dicte pas la façon dont le score est calculé [Chang *et al.*, 2006]. La norme utilise aussi la durée et le nombre des tentatives faites par un apprenant pour atteindre un objectif donné.

Dans nos travaux de recherches, nous nous intéressons à l'évaluation adaptative, c'est-à-dire l'adaptation des ressources et de la stratégie d'apprentissage dans un processus d'enseignement en se basant sur l'évaluation de l'apprenant. Dans ce contexte, l'état de l'art que nous avons effectué détaille les méthodes d'adaptation existantes dans le domaine d'éducation. Cependant, nous avons révélé des faiblesses dans ces méthodes que nous détaillons par la suite.

Dans l'enseignement d'un langage de programmation, un système d'adaptation a été développé pour améliorer l'apprentissage conçue autour du modèle domaine, un ensemble de sujets, un type de contenu, qui est difficile à réutiliser dans un autre contexte [Sosnovsky et Brusilovsky, 2015]. Dans un contexte du monde réel, le processus d'adaptation basé sur les Taxonomies de Bloom et d'ACM a un effet de chemin mais pas d'ajustement d'adaptation en cas d'échec, qui est pris en compte dans notre approche.

Par ailleurs, une navigation adaptative basée sur le score d'un apprenant a été conçue dans les Systèmes de Gestion de l'Apprentissage (SGA), mais le modèle d'évaluation (score) n'est pas intuitif et ne fait pas usage de traces pour l'adaptation.

Dans un contexte différent du nôtre, les jeux éducatifs, deux types d'adaptation ont été proposés, la macro et micro adaptation qui utilisent des scénarios prédéfinis. Cependant, notre approche n'a pas besoin de prédéfinir une séquence de situations. Ils sont définis dynamiquement en fonction du niveau de l'apprenant et de son acquisition de connaissances.

Elle pourrait être considérée comme une instanciation axée sur les traces de l'apprentissage pour l'adaptation, où les processus d'apprentissage sont dynamiquement adaptés et les parcours d'apprentissage sont également dynamiques et personnalisés pour chaque apprenant afin qu'il puisse passer une autre situation pédagogique en fonction de sa performance, et qu'il soit évalué dans notre cas en utilisant une fonction de notation basée sur les traces.

L'architecture de formation et d'apprentissage TLA a permis une traçabilité dans les plateformes d'apprentissage que SCORM ne pourrait pas facilement utiliser et capturer les données des apprenants et leurs expériences d'apprentissage. Cependant, dans notre travail, nous nous sommes concentrés sur les modèles de traces des apprenants pour l'adaptation de la stratégie de formation, considérée comme un travail de recherche futur pour xAPI comme indiqué dans [Hruska *et al.*, 2014].

Enfin, une combinaison d'IMS-LD avec une ontologie du domaine a eu lieu pour avoir plus de sémantique des spécifications de la norme [Burgos *et al.*, 2007]. Cependant, il ne tient pas compte de la modélisation des traces et des concepts d'évaluation des apprenants, qui sont mis en œuvre dans notre modèle d'ontologie.

Dans les Systèmes Tutoriels Intelligents (STI), seuls quelques travaux ont abordé l'évaluation entièrement intégrée. Les méthodes d'évaluations proposées dans ces systèmes sont classiques et statiques.

Notre approche couvre trois types d'évaluations : diagnostique, formative et sommative à base des traces temporelles et le nombre de tentatives qui sont utiles pour le processus de diagnostic afin d'évaluer le niveau de connaissances des apprenants. L'approche proposée peut également être utilisée au cours du processus de formation fournissant une évaluation formative ou une évaluation sommative après une session de formation.

## 2.8 Conclusion

Dans ce chapitre 2, nous avons détaillé l'évolution des méthodes d'évaluation de l'apprenant de l'évaluation traditionnelle faite par un enseignant ou un formateur à l'évaluation automatisée faite par un logiciel. Nous nous sommes intéressés aux méthodes de notation automatisées dans les systèmes d'apprentissage en ligne. Dans ces derniers, les réponses considérées peuvent être ouvertes (réponses courtes ou essais), fermées (vrai/faux ou multiples) ou orales. Ces réponses sont évaluées par des techniques de notation automatisée telles que l'évaluation automatisée des essais, gestion d'admission des diplômés, examen de diplôme, c-Rater, etc. Nous avons étudié l'intégration de ces évaluateurs automatisés dans les systèmes tutoriels intelligents (STI). Ensuite, nous avons recensé quelques méthodes d'évaluation adaptées comme les Tests Informatiques Adaptatifs (TIA) qui permettent une adaptation des questions suivant la performance de l'apprenant.



Par la suite, nous avons détaillé les systèmes d'adaptation existants dans les plateformes d'apprentissage en ligne. Les systèmes d'adaptation proposés permettent une adaptation de la ressource ou du trajectoire d'apprentissage. Enfin, nous avons étudié l'adaptation des systèmes tutoriels intelligents qui sont notre domaine d'application.

Dans les systèmes d'évaluation actuels, quand la réponse est correcte, le système mentionne que l'apprenant a acquis la connaissance. Dans notre cas, même si la réponse est correcte, nous pouvons savoir, à travers les traces d'une activité, le niveau d'acquisition des connaissances (faible, moyen ou excellent). Par exemple, si un apprenant répète cinq fois le même exercice pour le résoudre et/ou met deux heures pour avoir une réponse correcte, cela montre qu'il y a un problème dans la stratégie d'apprentissage. Par conséquent, nous avons besoin d'extraire d'autres données nécessaires pour évaluer les apprenants et détecter les faiblesses d'une stratégie d'apprentissage. Cette évaluation à base de traces nous permet une adaptation de la ressource ou du trajectoire d'apprentissage de l'apprenant.

Nous présentons dans les chapitres qui suivent la démarche suivie pour la mise en place de nos modèles d'évaluation fondés sur les traces qui sont la base de notre modèle d'adaptation de la ressource et la stratégie d'apprentissage dans un (STI).

## **Partie III**

# **Évaluation et Adaptation de la stratégie d'apprentissage pour l'apprenant à base de traces**



# Table des matières

---

<b>Chapitre 3 Modèles d'évaluation à base de traces</b>	<b>84</b>
3.1 Introduction . . . . .	84
3.2 Concepts et définitions . . . . .	85
3.3 Modèle d'évaluation à base de traces . . . . .	88
3.4 Discussion et conclusion . . . . .	95
<b>Chapitre 4 Modèle d'adaptation de ressources et de stratégies d'apprentissage à base de traces</b>	<b>96</b>
4.1 Introduction . . . . .	96
4.2 Processus d'adaptation à base de traces . . . . .	97
4.3 Ontologie d'adaptation du domaine de formation à base de traces . . . . .	98
4.4 Adaptation de la stratégie d'apprentissage à base de traces . . . . .	101
4.5 Discussion et conclusion . . . . .	105

---

L'utilisation de cours en ligne s'est accrue dans différents domaines où la formation est nécessaire pour faire progresser les connaissances et les compétences techniques des apprenants [Gill, 2011; Snyder *et al.*, 2016]. Bien que les changements récents vers des programmes à grande échelle soient prometteurs, ils posent des défis à l'évaluation, en particulier l'évaluation des compétences des apprenants. Ces évaluations font partie intégrante de la plupart des programmes éducatifs et sont souvent utilisées pour comprendre les trajectoires des compétences des individus [Biggs, 2011]. Les systèmes d'évaluation automatisés offrent un moyen possible d'évaluer les compétences des apprenants dans les apprentissages en ligne [Williamson *et al.*, 2012].

L'intégration des systèmes d'évaluation dans l'apprentissage en ligne est important dans la personnalisation et l'adaptation du contenu et la création de profils utilisateurs en fonction de leurs comportements d'apprentissage. Cela augmente l'efficacité des plateformes d'apprentissage en ligne, obtenant ainsi une grande acceptation car elles permettent aux apprenants de définir leurs propres objectifs, d'apprendre à leur propre rythme, et même de décider de leur méthode d'apprentissage, ce qui conduit à de meilleurs résultats d'apprentissage. Afin de personnaliser le processus d'apprentissage et d'adapter le contenu à chaque apprenant, les systèmes d'apprentissage en ligne peuvent utiliser différentes stratégies.

Dans cette partie du manuscrit, nous présentons nos contributions concernant l'évaluation des apprenants dans les systèmes tutoriels intelligents à base de traces produites pendant une activité d'apprentissage. Notre approche d'évaluation fondée sur les traces nous a permis de construire notre système d'adaptation à base de traces de la ressource et/ou de la situation d'apprentissage selon le niveau d'acquisition des connaissances de l'apprenant.

Cette partie est constituée de deux chapitres. Dans le chapitre 3, nous présentons les trois modèles d'évaluation proposés, modèle d'évaluation fondé sur la durée d'interaction, modèle d'évaluation fondé sur le nombre de tentatives et modèle d'évaluation combiné fondé sur la durée et le nombre de tentatives. Le chapitre 4 est consacré au système d'adaptation de ressources et de la stratégie d'apprentissage à base de traces dans un système STI.

# Chapitre 3

## Modèles d'évaluation à base de traces

### 3.1 Introduction

Au cours des dernières années, l'utilisation de plateformes d'apprentissage en ligne s'est répandue considérablement. Ces systèmes ont adopté les NTIC dans leurs processus d'enseignement [Martin, 2008] pour assister l'apprenant avec des outils et des ressources durant son acquisition des connaissances dans les activités d'apprentissage telles que des cours, des exercices, des jeux-questionnaires, etc., qui sont organisées de telle sorte que les apprenants peuvent gérer et contrôler les activités d'apprentissage. Afin d'être en mesure de suivre leur progrès en termes d'acquisition de connaissances, les apprenants doivent passer des tests et des examens sous forme de questions à choix multiples et/ou des exercices de complétion (fill-in-the-blank) [Nicol, 2007; Crisp, 2009], qui retournent une évaluation statique basée sur les réponses des apprenants. Ces méthodes d'évaluation, dans le sens des fonctions de classement ou de notation [Crooks, 1988], font parties d'un composant que l'on appelle e-évaluation intégré dans quelques plateformes d'apprentissage.

Le processus d'évaluation fondé sur les traces permet de calculer des indicateurs pertinents qui permettent la personnalisation de l'analyse [Settouti *et al.*, 2009c], étant donné que les traces peuvent être rattachées à un utilisateur unique. Cette personnalisation nous pouvons obtenir plus de précision dans l'analyse et la compréhension du comportement de l'apprenant. Par exemple, un apprenant qui a mis deux fois plus de temps que recommandé pour résoudre un type de problème peut révéler quelques difficultés pour acquérir les connaissances requises. La raison pourrait être liée aux déficiences des apprenants en pré-requis et/ou à l'inadéquation de la stratégie de formation (ressources et processus de formation). Par conséquent, les traces peuvent divulguer et révéler de nouvelles connaissances sur les apprenants et les stratégies d'enseignement qui sont en fait difficiles à atteindre avec les méthodes d'évaluation traditionnelles. De plus, les traces sur le nombre d'essais ou tentatives réalisés par les apprenants pour résoudre

un problème peuvent, par exemple, informer les experts à la fois du niveau de connaissances maîtrisés et l'adéquation des ressources d'enseignement aux profils des apprenants.

Nous présentons dans ce chapitre un modèle d'évaluation de l'apprenant basé sur deux types de trace, à savoir, (i) le temps passé pour résoudre un problème, (ii) le nombre de tentatives pour le résoudre. Leur combinaison est également prise en compte. Concrètement, nous proposons trois modèles qui calculent les indicateurs de performance de l'apprenant et les scores raffinés pour le niveau de connaissances acquises par l'apprenant.

L'approche proposée est une première étape vers un modèle de comportement basé sur la traçabilité des apprenants pour l'acquisition des connaissances et l'adaptation des ressources de formation. L'importance de notre approche est double :

- le calcul des indicateurs de performance personnalisés pour les ressources de formation et processus d'adaptation. Par exemple, si le système constate que l'apprenant échoue fréquemment pour résoudre un type de problème, alors il recommande certaines actions à effectuer pour adapter la ressource et/ou le processus de formation sur son acquisition de connaissances ;
- l'approche proposée peut être considérée comme une méthode de notation et de classement. En effet, l'approche introduite peut distinguer entre les apprenants ayant répondu correctement à une activité de test donnée mais dans des durées et nombre de tentatives différents.

Le reste de ce chapitre est structuré comme suit. La section 3.2 donne des définitions de traces et leur gestion et exploitation dans un Système à Base de Trace (SBT), ensuite nous présentons des fonctions classiques d'évaluation quantitative de l'apprenant. La section 3.3 détaille notre modèle d'évaluation à base de traces. Dans la section 3.4, nous clôturons le chapitre avec une conclusion.

## 3.2 Concepts et définitions

### 3.2.1 Définition des Traces

Dans la littérature, la notion de trace se réfère souvent aux fichiers journaux (*logs*) décrivant les événements survenus dans un système donné [Andrews, 1998]. Cependant, dans le monde réel, une trace est définie comme suit.

**Définition 3.2.1.** Une trace est une série de choses laissées par toute action liée à un être ou à un objet. Une série d'empreintes ou de marques laissées par le passage d'un objet ou d'un être, qui révèle que quelque chose a existé [Djouad *et al.*, 2010]. De ce fait, dans le domaine d'apprentissage en ligne, une trace est définie comme suit.

Dans notre contexte, une trace numérique est toute donnée capturée par des processus d'observation durant une activité d'apprentissage. Elle représente une empreinte résiduelle des interactions entre les apprenants et les processus de formation.

**Définition 3.2.2.** Une trace numérique d'une activité consiste en plusieurs objets qui sont chronologiquement observés, capturés et sauvegardés sur un support.

Ces traces sont collectées et analysées par un Système à Base de Trace (SBT) [Settoui *et al.*, 2009a; Laflaquiere *et al.*, 2006]. Dans [Mille *et al.*, 2013], un SBT est décrit par trois éléments :

- **Observé** : agent système qui est attaché à un événement survenu lors de l'exécution d'une application. Chaque observé est caractérisé par un nom, un horodatage et un ensemble de propriétés.
- **Modèle de trace** : description abstraite des éléments d'une trace. En d'autres termes, il représente un vocabulaire de traces dont chacune peut être associée à un modèle.
- **M-trace (pour trace modélisée)** : une trace associée à son modèle et gérée par un système à base de m-trace (m-SBT).

La figure 3.1 illustre deux exemples d'observés : un utilisateur et une situation de formation et leurs traces décrites respectivement par  $\{identifiant\ utilisateur (UserID), Date\}$  et  $\{Nom (Name), Durée (TimeResp), Score, Tentatives (Attempts)\}$ .

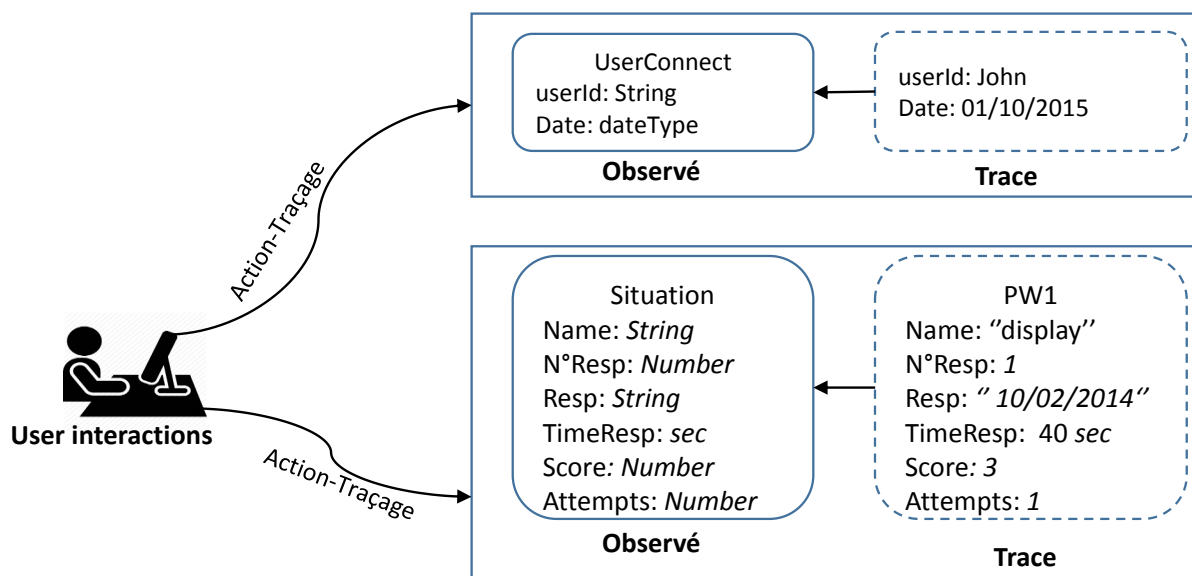


FIGURE 3.1 – Aperçu du système à base de traces.



### 3.2.2 Définition de l'évaluation

Dans notre cas, nous définissons une fonction d'évaluation quantitative qui calcule une valeur dans  $\mathbb{R}$ .

**Définition 3.2.3** (Fonction Quantitative de Notation). Une fonction quantitative de notation est toute fonction  $f$  définie de  $2^{\mathcal{A}}$ , telle que  $\mathcal{A}$  est l'ensemble de toutes les réponses correctes dans  $[0, M]$ , telle que  $M \in \mathbb{N}$  est la note maximale et satisfait les propriétés suivantes :

- $f$  est une fonction équitable : dans le sens où  $f$  est une fonction non décroissante sur l'ensemble  $2^{\mathcal{A}}$ ,
- $f$  est une fonction bornée :  $f(\emptyset) = 0$  et  $f(\mathcal{A}) = M$ ,
- $f$  est minimale dans le sens où pour tout ensemble de réponses  $\mathcal{B}$  contenant  $\mathcal{A}$ ,  $f(\mathcal{B}) < f(\mathcal{A})$ .

Nous étendons la définition de la fonction quantitative de notation aux traces pour définir la fonction quantitative de notation à base de traces.

**Définition 3.2.4** (Fonction Quantitative d'Evaluation à Base de Traces). Soit  $f$  une fonction de notation quantitative définie de  $2^{\mathcal{A}}$  dans  $[0, M]$  et  $T$  une trace définie sur un domaine  $\mathbb{D}$ . Une fonction de notation quantitative à base de traces est une fonction binaire  $g$  définie de  $2^{\mathcal{A}} \times \mathbb{D}$  dans  $[0, M]$  comme suit.

$$\begin{aligned} g : 2^{\mathcal{A}} \times \mathbb{D} &\longrightarrow [0, M] \\ (A, d) &\longmapsto g(f(A), d) \end{aligned}$$

Telle que  $g$  est bornée, limitée et minimale.

**Définition 3.2.5** (Scénario et situation). Nous définissons un scénario  $\mathcal{S}$  comme une suite de situations  $\mathcal{S} = \{S_0, \dots, S_{i-1}, S_i, S_{i+1}, \dots, S_n\}$ . Chaque situation  $S_i$  est un ensemble de ressources ordonnées par leur niveau de difficulté noté  $S_i = \{R_1 \prec R_2 \prec \dots \prec R_m\}$ , où  $R_j$  avec  $j \in \{1, \dots, m\}$  est une ressource.

Sur la base des définitions susmentionnées dans la section 3.2, nous définissons dans ce qui suit nos modèles d'évaluation basés sur les traces d'apprentissage. Les traces considérées dans ces modèles sont la durée d'interaction et le nombre de tentatives effectués par un apprenant durant son apprentissage.

### 3.3 Modèle d'évaluation à base de traces

Dans cette section, nous détaillons nos modèles d'évaluation fondés sur les traces d'apprentissage [Chachoua *et al.*, 2016d]. Dans la sous-section 3.3.1, nous détaillons le modèle à base de traces temporelles qui prend en compte la durée d'interaction. Dans la sous-section 3.3.2, notre second modèle basé sur le nombre de tentatives est décrit. La combinaison des deux traces est introduite dans la sous-section 3.3.3.

#### 3.3.1 Modèle d'évaluation fondé sur la durée d'interaction

Une activité d'évaluation d'apprentissage peut lui être associée deux horodatages correspondant respectivement à son instant de début et à son instant de fin. Ces horodatages renvoient la durée que l'apprenant a consacré à l'activité considérée. Le temps passé par l'apprenant est utile pour son évaluation. L'intuition derrière notre modèle d'évaluation basé sur la trace temporelle est comme suit :

- supposons qu'un objectif de formation est modélisé à l'aide d'un ensemble de  $m$  activités de test  $O = \{A_1, \dots, A_m\}$ . Une activité de test peut être un exercice de remplissage des blancs, exercice de réorganisation, une valeur à calculer pour résoudre un problème, une question à choix unique ou multiple, etc. ;
- chaque activité ou exercice  $A_i$  avec  $i = \{1, \dots, m\}$  est caractérisée par une note maximale, notée  $\mathcal{M}_{A_i}$  qui peut être définie par un expert et/ou un formateur dans le système d'apprentissage en ligne et une durée optimale qui peut aussi être fixée par un expert et/ou un formateur, notée  $\mathcal{D}_{A_i}$  ;
- à partir des traces temporelles, il est possible de calculer la durée prise par un apprenant pour répondre au problème. Dénotons cette durée par  $d$  ;
- si la réponse de l'apprenant est fautive, alors sa note est 0 ;
- si l'apprenant répond juste à l'exercice en une durée inférieure ou égale à la durée optimale  $\mathcal{D}_{A_i}$ , alors il obtient la note complète  $\mathcal{M}_{A_i}$  ;
- si l'apprenant a pris plus de temps que la durée optimale, alors plus le temps supplémentaire pris pour répondre correctement à la question est élevé, plus la note calculée diminue.

La note obtenue par l'apprenant pour l'activité de test  $A_i$  est calculée par la fonction binaire  $M_{A_i}(\mathcal{M}_{A_i}, d)$  (introduite dans la définition 3.2.4), et définie par le modèle (3.1) sur la base de la durée prise pour répondre à un exercice ou une activité de test :

$$M_{A_i}(\mathcal{M}_{A_i}, d) = \begin{cases} \mathcal{M}_{A_i} & \text{si } (d \leq \mathcal{D}_{A_i}), \\ \mathcal{M}_{A_i} e^{-\alpha(\frac{d-\mathcal{D}_{A_i}}{\mathcal{D}_{A_i}})} & \text{sinon.} \end{cases} \quad (3.1)$$

tel que :

- $\alpha \in [0, 1]$  est une constante d'atténuation. Elle permet de calculer une note dans  $[0, \mathcal{M}_{A_i}]$ , ce qui indique que  $M_{A_i}(\mathcal{M}_{A_i}, d)$  est bornée. Si  $\alpha$  est proche de 0, alors  $M_{A_i}(d)$  approche  $\mathcal{M}_{A_i}$ . Sinon, si  $\alpha$  approche 1 alors  $M_{A_i}(\mathcal{M}_{A_i}, d)$  s'approche de 0. Plus  $\alpha$  se rapproche de 1, plus l'atténuation sera sévère. En outre, pour une valeur fixe pour le paramètre  $\alpha$ , plus la durée  $d$  est longue, plus la note calculée est faible, ce qui montre l'équité de la fonction binaire  $M_{A_i}(\mathcal{M}_{A_i}, d)$ ,
- La fraction  $(\frac{d-\mathcal{D}_{A_i}}{\mathcal{D}_{A_i}})$  vise à normaliser la durée pour éliminer l'effet de son unité de mesure sur la note calculée.

Il est intéressant de noter que le terme exponentiel paramétré dans le modèle (3.1) permet de capturer l'effet naturel d'atténuation de la note calculée lorsque la durée supplémentaire augmente. Par conséquent, on en déduit que ce modèle possède la capacité d'exprimer l'importance de la compétence que l'apprenant doit acquérir dans une activité de formation donnée en configurant le paramètre constant  $\alpha$ . En effet, plus la compétence est importante, plus proche de 1 la constante  $\alpha$  devrait être. Cette flexibilité n'est pas facile à mettre en œuvre avec un modèle linéaire. De plus, lorsque  $\alpha = 0$  alors le modèle (3.1) se réduit à une fonction de notation traditionnelle, qui renvoie la note complète si la réponse est correcte indépendamment des traces temporelles.

**Exemple 3.3.1.** Supposons un système d'apprentissage en ligne pour la formation SQL. Supposons un exercice sur une requête SQL pour laquelle nous définissons les paramètres suivants : (i)  $\mathcal{M}_{A_i} = 5$ , (ii)  $\mathcal{D}_{A_i} = 300$  Secondes. Pour illustrer le comportement du modèle (3.1), nous l'avons exécuté trois fois avec trois valeurs différentes de  $\alpha$  telles que :

- $\alpha = 0,2$  (activité de faible niveau d'importance) ;
- $\alpha = 0,5$  (activité de niveau d'importance moyen) ;
- $\alpha = 0,8$  (activité de niveau d'importance élevé).

La table 3.3.1 résume les résultats obtenus pour 4 apprenants  $\{a_1, a_2, a_3, a_4\}$  avec leurs durées diverses prises pour résoudre l'exercice.

TABLE 3.3.1 – Exemple d'instanciation du modèle basé sur la trace temporelle.

Apprenant	$M_{A_i}(\mathcal{M}_{A_i}, d)$			
	$d(s)$	$\alpha = 0,2$	$\alpha = 0,5$	$\alpha = 0,8$
$a_1$	200	5,00	5,00	5,00
$a_2$	300	5,00	5,00	5,00
$a_3$	530	4,29	3,41	2,71
$a_4$	620	4,04	2,93	2,13

Nous notons que les notes diminuent en fonction du temps supplémentaire pris par l'apprenant pour accomplir une activité. Avec  $\alpha = 0,2$  la note de l'apprenant  $a_4$  a diminué d'environ 20% par rapport à la note complète, même s'il a pris plus de deux fois plus de temps que recommandé et de plus de 57% lorsque  $\alpha = 0,8$ .

### 3.3.2 Modèle d'évaluation fondé sur le nombre de tentatives

Dans cette partie, nous définissons notre deuxième modèle de notation basé sur le nombre de tentatives. Comme mentionné dans l'introduction, le nombre de tentatives faites par l'apprenant est également utile pour son évaluation. L'intuition derrière le modèle est la suivante :

- chaque exercice ou activité  $A_i$  avec  $i = \{1, \dots, m\}$  est caractérisée par une note maximale, notée  $\mathcal{M}_{A_i}$  qui peut être définie par un expert et/ou un formateur dans le système d'apprentissage en ligne et un nombre de tentatives autorisées, noté  $\mathcal{T}_{A_i}$  ;
- à partir de la trace du nombre de tentatives, il est possible de calculer le nombre d'essais d'un apprenant donné pour résoudre le problème ; désignons ce nombre de tentatives par  $t$  ;
- si la réponse de l'apprenant est fautive, alors sa note est 0 ;
- si l'apprenant répond bien à l'exercice avec un nombre d'essais inférieurs ou égal à  $\mathcal{T}_{A_i}$ , alors il obtient la note complète  $\mathcal{M}_{A_i}$  ;
- si l'apprenant fait plus de tentatives que  $\mathcal{T}_{A_i}$ , alors plus le nombre de tentatives réalisés par l'apprenant pour répondre à la question est élevé, plus la note calculée est plus basse.

La note obtenue par l'apprenant pour l'activité de test  $A_i$  est formalisée par le modèle de notation  $M_{A_i}(\mathcal{M}_{A_i}, t)$  (telle qu'elle est introduite dans la définition 3.2.4) définie par le modèle (3.2) sur la base du nombre de tentatives  $t$  :

$$M_{A_i}(\mathcal{M}_{A_i}, t) = \begin{cases} \mathcal{M}_{A_i} & \text{si } (t \leq \mathcal{T}_{A_i}), \\ \mathcal{M}_{A_i} e^{-\beta \left(\frac{t - \mathcal{T}_{A_i}}{\mathcal{T}_{A_i}}\right)} & \text{sinon.} \end{cases} \quad (3.2)$$

tel que :

- $\beta \in [0, 1]$  : est une constante d'atténuation. Elle permet de calculer une note dans  $[0, \mathcal{M}_{A_i}]$ , donc  $M_{A_i}(\mathcal{M}_{A_i}, t)$  est bornée. Si  $\beta$  est proche de 0, alors  $M_{A_i}(\mathcal{M}_{A_i}, t)$  s'approche de  $\mathcal{M}_{A_i}$ . Dans le cas contraire, si  $\beta$  approche 1, alors  $M_{A_i}(\mathcal{M}_{A_i}, t)$  s'approche de 0. Plus  $\beta$  se rapproche de 1, plus sévère l'atténuation sera. De plus, pour un paramètre fixe  $\beta$ , plus le nombre de tentatives  $t$  est élevé, plus la note calculée est faible. Par conséquent, la fonction  $M_{A_i}(\mathcal{M}_{A_i}, t)$  est équitable ;
- La fraction  $\left(\frac{t - \mathcal{T}_{A_i}}{\mathcal{T}_{A_i}}\right)$  normalise le nombre d'essais pour la note calculée par la fonction  $M_{A_i}(\mathcal{M}_{A_i}, t)$ .

Nous notons que les modèles (3.1) et (3.2) partagent des propriétés communes. La formule (3.2) est flexible (par le terme exponentiel) pour capturer l'effet d'atténuation et exprimer l'importance de l'activité de formation  $A_i$  et se comporte comme une fonction de notation traditionnelle quand  $\beta = 0$ .

### Exemple 3.3.2.

- $\beta = 0,2$  (activité de faible niveau d'importance) ;
- $\beta = 0,5$  (activité de niveau moyen d'importance) ;
- $\beta = 0,8$  (activité de niveau élevé d'importance).

Considérons le même exercice de l'exemple 3.3.1 pour lequel les paramètres du modèle sont définis comme suit : (i)  $\mathcal{M}_{A_i} = 5$ , et (ii) le nombre d'essais  $\mathcal{T}_{A_i} = 3$ . Pour illustrer le comportement du modèle (3.2), nous considérons 4 apprenants  $\{a_1, a_2, a_3, a_4\}$  et leurs nombre respectif de tentatives pour résoudre l'exercice. Les valeur de  $\beta$  sont :

- $\beta = 0,2$  (activité de faible niveau d'importance) ;
- $\beta = 0,5$  (activité de niveau moyen d'importance) ;
- $\beta = 0,8$  (activité de niveau élevé d'importance).

TABLE 3.3.2 – Exemple d'instanciation du modèle basé sur la trace de tentative.

Apprenant	$M_{A_i}(\mathcal{M}_{A_i}, t)$			
	Tentatives $t$	$\beta = 0,2$	$\beta = 0,5$	$\beta = 0,8$
$a_1$	1	5,00	5,00	5,00
$a_2$	4	4,68	4,23	3,83
$a_3$	2	5,00	5,00	5,00
$a_4$	6	4,09	3,03	2,25

Nous remarquons dans le tableau 3.3.2 le même effet d'atténuation dans les résultats et la sévérité de la fonction de notation lorsque  $\beta$  augmente (c'est-à-dire se rapproche de 1).

### 3.3.3 Modèle d'évaluation combiné fondé sur la durée et le nombre de tentatives

En se basant sur les deux modèles précédents, nous définissons un modèle combiné basé sur la durée et le nombre de tentatives. Nous rappelons qu'une activité  $A_i$  a une note complète  $\mathcal{M}_{A_i}$ , qui peut être définie par un expert et/ou un formateur dans le système, une durée optimale  $\mathcal{D}_{A_i}$  et un nombre autorisé de tentatives  $\mathcal{T}_{A_i}$ . Notons la durée prise par un apprenant et son nombre de tentatives par  $d$  et  $t$  respectivement. Par conséquent, l'évaluation de l'apprenant, notée  $M_{A_i}(\mathcal{M}_{A_i}, d, t)$ , selon ces paramètres est la suivante :

- si  $(d \leq \mathcal{D}_{A_i})$  et  $(t \leq \mathcal{T}_{A_i})$  alors  $M_{A_i}(\mathcal{M}_{A_i}, d, t) = \mathcal{M}_{A_i}$  (la note complète) ;
- si  $(d > \mathcal{D}_{A_i})$  et  $(t \leq \mathcal{T}_{A_i})$  alors  $M_{A_i}(\mathcal{M}_{A_i}, d, t) = M_{A_i}(\mathcal{M}_{A_i}, d)$  (en utilisant le modèle (3.1)) ;
- si  $(d \leq \mathcal{D}_{A_i})$  et  $(t > \mathcal{T}_{A_i})$  alors  $M_{A_i}(\mathcal{M}_{A_i}, d, t) = M_{A_i}(\mathcal{M}_{A_i}, t)$  (en utilisant le modèle (3.2)) ;
- si  $(d > \mathcal{D}_{A_i})$  et  $(t > \mathcal{T}_{A_i})$  alors nous agrégeons intuitivement les deux traces en additionnant le temps et le nombre supplémentaire de tentatives, puis nous appliquons le même raisonnement que pour les modèles (3.1) et (3.2) pour définir la fonction d'évaluation de trois paramètres  $M_{A_i}(\mathcal{M}_{A_i}, d, t)$ . Par conséquent, le modèle combiné peut être exprimé par le modèle (3.3) comme suit :

$$M_{A_i}(\mathcal{M}_{A_i}, d, t) = \begin{cases} \mathcal{M}_{A_i} & \text{si } (d \leq \mathcal{D}_{A_i}) \wedge (t \leq \mathcal{T}_{A_i}) \\ M_{A_i}(\mathcal{M}_{A_i}, d) & \text{si } (d > \mathcal{D}_{A_i}) \wedge (t \leq \mathcal{T}_{A_i}) \\ M_{A_i}(\mathcal{M}_{A_i}, t) & \text{si } (d \leq \mathcal{D}_{A_i}) \wedge (t > \mathcal{T}_{A_i}) \\ \mathcal{M}_{A_i} e^{-\gamma(\frac{t-\mathcal{T}_{A_i}}{\mathcal{T}_{A_i}} + \frac{d-\mathcal{D}_{A_i}}{\mathcal{D}_{A_i}})} & \text{sinon.} \end{cases} \quad (3.3)$$

tel que :

- $\gamma \in [0, 1]$  est une constante d'atténuation. Elle permet de calculer une note dans  $[0, \mathcal{M}_{A_i}]$ , donc la fonction  $M_{A_i}(\mathcal{M}_{A_i}, d, t)$  est bornée. Si  $\gamma$  est proche de 0, alors  $M_{A_i}(\mathcal{M}_{A_i}, d, t)$  s'approche de  $\mathcal{M}_{A_i}$ . Sinon, si  $\gamma$  s'approche de 1 alors  $M_{A_i}(\mathcal{M}_{A_i}, d, t)$  s'approche de 0. Plus  $\gamma$  se rapprochent de 1, plus l'atténuation sera sévère. De plus, pour un paramètre  $\gamma$  fixe, plus la durée  $d$  est importante et plus le nombre de tentatives  $t$  est élevé, plus la note calculée est faible. Par conséquent, la fonction  $M_{A_i}(\mathcal{M}_{A_i}, d, t)$  est équitable.

La modèle (3.3) a les mêmes propriétés que les modèles (3.1) et (3.2) en termes d'effet d'atténuation, de flexibilité et de notation régulière.

**Exemple 3.3.3.** Considérons les mêmes apprenants des deux exemples 3.3.1 et 3.3.2. Nous rappelons que les paramètres d'activité d'apprentissage sont les suivants : la note complète  $\mathcal{M}_{A_i} = 5$  points, la durée optimale  $\mathcal{D}_{A_i} = 300$  secondes et le nombre d'essais autorisés est  $\mathcal{T}_{A_i} = 3$ . Pour illustrer le comportement du modèle (3.3), nous considérons les mêmes traces obtenues pour les apprenants  $\{a_1, a_2, a_3, a_4\}$ . La table 3.3.3 résume les résultats obtenus pour les quatre apprenants sous les valeurs de  $\gamma$  suivantes :

- $\gamma = 0,2$  (activité de faible niveau d'importance);  $\gamma = 0,5$  (activité de niveau moyen d'importance);  $\gamma = 0,8$  (activité de niveau élevé d'importance).

TABLE 3.3.3 – Exemple d'instanciation du modèle combiné de la trace temporelle et du nombre de tentatives.

Apprenant	$M_{A_i}(\mathcal{M}_{A_i}, d, t)$				
	$d$	$t$	$\gamma = 0,2$	$\gamma = 0,5$	$\gamma = 0,8$
$a_1$	200	1	5,00	5,00	5,00
$a_2$	300	4	4,68	4,23	3,83
$a_3$	530	2	4,29	3,41	2,71
$a_4$	620	6	3,31	1,78	0,96

Nous remarquons dans le tableau 3.3.3 le double effet de sanction pour l'apprenant  $a_4$  en raison du temps supplémentaire pris pour résoudre le problème et du nombre d'essais qui excède le double de ce qui est autorisé.

### 3.3.4 Processus d'évaluation automatisée à base de traces

La Figure 3.2 décrit les composants de notre processus de notation automatisé à base de traces composé des éléments suivants :

- un système d'apprentissage en ligne équipé d'une interface utilisateur pour interagir avec les apprenants ;
- un module d'évaluation qui met en œuvre les fonctions de notation et d'évaluation et fournit les résultats aux apprenants et enseignants/experts ;
- une base de données m-trace est alimentée par le système d'apprentissage avec les traces des utilisateurs. Elle fournit au système d'évaluation des traces enregistrées.

Les étapes du processus d'évaluation sont également décrites comme suit :

Les apprenants interagissent avec le système d'apprentissage en ligne dans le but de réaliser une activité de formation. Leurs actions et interactions sont suivies et les traces sont enregistrées dans la base de traces. Ces dernières sont ensuite modélisées selon les trois phases suivantes [Berkani et Bellatreche, 2017] :

- la collecte des données ;
- la transformation des données ;
- l'analyse des données [Settouti *et al.*, 2009b].

Ensuite, une phase de test est lancée pour vérifier si la réponse est correcte ou fautive [Patelis, 2000; Thompson et Wiess, 2009; Sireci et Luecht, 2012]. Après cela, nous appliquons un de nos modèles d'évaluation :

1. le modèle à base de trace temporelle ;
2. le modèle à base de la trace de tentatives ;
3. le modèle combiné basé sur la trace temporelle et le nombre de tentatives pour la progression des apprenants et la stratégie d'apprentissage.



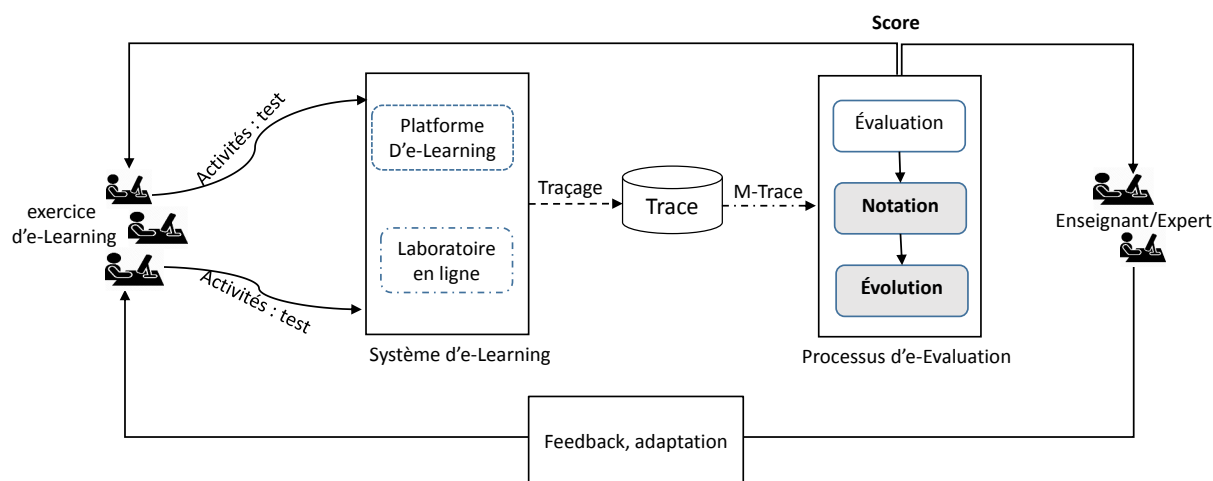


FIGURE 3.2 – Aperçu du déroulement du processus d'évaluation.

### 3.4 Discussion et conclusion

Nous avons développé des modèles d'évaluation des apprenants pendant une activité d'apprentissage, basés sur la durée, le nombre de tentatives, et leur combinaison, qui sont considérés ici comme des traces modélisées. Les modèles proposés évaluent la performance de l'acquisition de connaissances des apprenants au cours d'une activité de formation.

En outre, les experts peuvent configurer dans le modèle l'importance d'une activité de formation, ou limiter le modèle pour effectuer une évaluation régulière en réglant les constantes d'atténuation à 0. Ce type d'évaluation permet la personnalisation et l'adaptation de la stratégie de formation suivie par l'apprenant et une évaluation personnalisée de l'acquisition de connaissances.

Toutefois, l'inconvénient principal de nos modèles d'évaluation à base de traces réside dans la difficulté de calculer les paramètres d'atténuation  $\alpha$ ,  $\beta$  et  $\gamma$  car elles sont fixées par un formateur et/ou un expert de formation. Un autre désavantage concernant ces paramètres est la possibilité de considérer un seul paramètre identique à tous les modèles, pour cela il est nécessaire d'avoir une expertise dans le domaine pour prévoir la possibilité de régler ces paramètres en fonction des apprenants. Par exemple, une atténuation forte décourage les apprenants et les pousse à abandonner la formation, etc. Donc, il est nécessaire de recourir à des expérimentations sur le terrain pour trouver l'équilibre idéal entre l'atténuation et l'évaluation constructive d'un apprenant.

Comme nous allons le décrire dans le chapitre 4, les modèles d'évaluation sont à la base de nos travaux sur l'adaptation des ressources et/ou des situations d'apprentissage.

# Chapitre 4

## Modèle d'adaptation de ressources et de stratégies d'apprentissage à base de traces

### 4.1 Introduction

Les systèmes adaptatifs d'apprentissage sont une sorte de *Systèmes Interactifs Adaptatifs (SIA)* conçus pour aider les enseignants et les experts à affiner les stratégies de formation afin de mieux atteindre les objectifs pédagogiques. Les *SIA* étendent les systèmes d'apprentissage en ligne avec la capacité d'adapter leurs exécutions en détectant et traitant les incompréhensions en interactions avec des apprenants [Brim et Beaudouin-Lafon, 1995; Pham *et al.*, 2011].

L'inconvénient de telles approches réside dans le fait que les acteurs du système interagissant pendant l'exécution d'une situation dans le système peuvent être confrontés à des incompréhensions lorsque les données internes d'au moins un acteur évoluent différemment des autres, après la même séquence d'interactions dans un contexte commun. Les caractéristiques ou préférences des apprenants sont des entrées que le système adaptatif doit acquérir avant tout processus d'adaptation. Ces données sont souvent statiques et doivent être continuellement mises à jour afin d'obtenir un contenu pédagogique valide et de bonnes stratégies d'apprentissage. Pour surmonter cet inconvénient, d'autres approches basées sur les traces sont apparues [Andrews, 1998; Djouad *et al.*, 2010]. Actuellement, les traces aident les systèmes d'apprentissage en ligne à apprendre continuellement sur le profil de l'apprenant et à évaluer son acquisition de connaissances, puis à adapter les ressources et les stratégies d'apprentissage en conséquence de manière transparente afin d'améliorer l'acquisition d'apprentissage des apprenants. De plus, le processus d'évaluation à base de traces permet de calculer des indicateurs pertinents permettant de personnaliser l'analyse [Chachoua *et al.*, 2016d; Settouti *et al.*, 2009c], car par la personnalisation nous pouvons obtenir plus de précision sur le modèle et l'évaluation de l'apprenant sur lesquels se base le processus d'adaptation.

Nous présentons dans ce chapitre un système adaptatif basé sur les traces. Compte tenu de la possibilité d'attacher la durée et le nombre de tentatives à une activité de formation, nous pouvons utiliser ces traces modélisées pour évaluer à la fois le niveau de l'apprenant et son progrès. Basé sur les traces et le modèle d'évaluation des apprenants, notre système construit un modèle d'état-transition dans lequel les états représentent des situations et les transitions expriment des étapes d'adaptation d'un état à un autre, prenant en considération les traces recueillies sur les activités des apprenants. Le processus d'adaptation génère un parcours d'apprentissage dynamique d'une situation à une autre en formant une stratégie pédagogique adaptée au niveau des connaissances de l'apprenant. Les transitions dans notre modèle adaptatif sont gouvernées par des règles logiques et un mécanisme de raisonnement pour maintenir la cohérence du système. Les règles logiques et le mécanisme de raisonnement sont mis en œuvre au sein d'une ontologie de domaine conçue pour décrire les principaux concepts de notre système.

Pour cela, nous avons créé un modèle d'adaptation à base de traces, ses composants sont définis dans la section 4.2. Le modèle d'adaptation exploite une ontologie pour la description des traces d'activités et des traces d'évaluations comme détaillé dans la section 4.3. Par conséquent, nous avons proposé deux types d'adaptations comme détaillé dans la section 4.4, à savoir :

1. Adaptation de la ressource d'apprentissage
2. Adaptation de la stratégie d'apprentissage

Pour ce faire, nous avons proposé un algorithme qui fonctionne à base de règles ontologiques et qui fournit une adaptation suivant les chemins obtenus de notre diagramme état-transition également développé. Enfin, nous concluons le chapitre dans la section 4.5.

## 4.2 Processus d'adaptation à base de traces

Les composantes principales de l'architecture du processus d'adaptation à base de traces, comme illustré sur la figure 4.1, sont les suivantes :

- **traces primaires** : elles sont des traces brutes obtenues au cours d'une situation d'apprentissage ; elles sont exprimées sous forme de logs dans un fichier texte, qui sont par la suite sauvegardées sous forme de traces modélisées dans une base de données (m-trace) ;
- **base de données m-trace** : permet de stocker et de gérer les traces modélisées des apprenants ;
- **situation** : définit les séquences d'interaction dans un scénario de formation pour atteindre un objectif. Une interaction est une action effectuée par un apprenant durant une situation d'apprentissage ;

- **situation et ressource BDD** : est une base de connaissances des situations et des ressources décrivant les activités d'apprentissage comme les contenus d'enseignement stockés dans une base de données, et une modélisation ontologique des concepts nécessaires en termes de classes et propriétés de notre domaine. L'ontologie permet d'exprimer des règles logiques sur lesquelles un raisonnement est construit pour adapter les ressources et les situations;
- **modèle d'évaluation** : il met en œuvre le modèle d'évaluation basé sur les traces pour évaluer les apprenants au cours de leurs activités d'apprentissage [Chachoua *et al.*, 2016d];
- **processeur de la stratégie** : il met en œuvre le processus de raisonnement sur l'ontologie tenant en compte les traces d'activités pour générer des situations et des ressources adaptées. Le processus reçoit comme entrées les traces (durée et nombre de tentatives) à partir de la base de données m-trace, la note calculée par un modèle d'évaluation et la ressource actuelle dans la situation actuelle, le résultat produit est la prochaine ressource et/ou la situation adaptées.

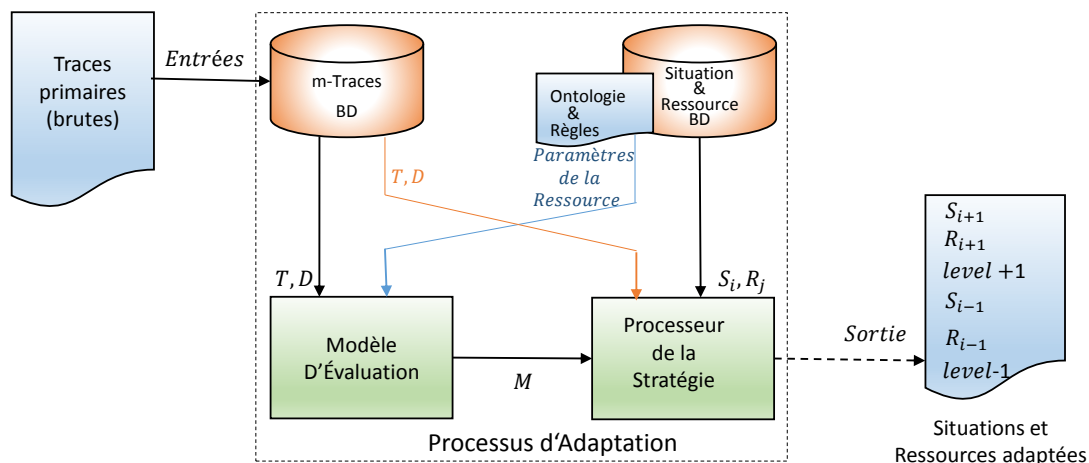


FIGURE 4.1 – Architecture du processus d'adaptation à base de traces.

### 4.3 Ontologie d'adaptation du domaine de formation à base de traces

Nous définissons notre modèle ontologique qui prend en considération la notion de situation [Pham *et al.*, 2013]. Figure 4.2 décrit le domaine ontologique de l'adaptation comme suit :

- **Situation** : peut être (mais sans s'y limiter) un cours, un quiz, un test, un travail pratique ou un travail dirigé, noté *CR*, *QZ*, *TS*, *TP* et *TD* respectivement, tel que :

- *LR* est un cours présenté par un enseignant ;
  - *DW* est le contenu d'apprentissage comme des exercices et des problèmes dédiés à approfondir les connaissances sur le sujet étudié dans un cours ;
  - *PW* est un contenu d'apprentissage pour expérimenter les connaissances acquises pendant une session de cours et/ou un travail dirigé ;
  - *TS* est un examen subi par les apprenants afin d'évaluer leur acquisition de connaissances ;
  - *QZ* est un ensemble de questions sur les connaissances ciblées dans *CR*, *TD* et *TP*.
- **Scenario** : est une succession d'exécution de situations jusqu'à atteindre un objectif global. Un scénario est conçu par un expert ou un enseignant ;
  - **Resource** : est un contenu d'apprentissage que les apprenants peuvent utiliser dans une situation donnée ;
  - **Learner** : est un acteur qui interagit avec les ressources dans une situation donnée de formation ;
  - **Mark** : est un processus qui évalue l'activité de l'apprenant en se basant sur les traces temporelles et le nombre de tentatives tel que détaillé dans [Chachoua *et al.*, 2016d] ;
  - **Trace Temporelle** : est la durée mise par un apprenant pour accomplir l'objectif de la situation ;
  - **Trace de Tentatives** : est le nombre de fois l'apprenant a refait la situation pour atteindre son objectif.

Les concepts définis ci-dessus sont liés par des propriétés d'objet pour décrire la base de connaissances. Cette dernière est basée sur des axiomes logiques qui permettent d'exprimer des règles ontologiques et des processus de raisonnement. Les axiomes identifiés de l'ontologie de la Figure 4.2 sont détaillés à l'aide de règles existentielles comme suit :

- $\forall x \text{Scenario}(x) \Rightarrow \exists y \text{Situation}(y) \wedge \text{isComposedOf}(x, y)$
- $\forall x \text{Situation}(x) \Rightarrow \exists y \text{trace}(y) \wedge \text{hasTrace}(x, y)$
- $\forall x \text{TimeTrace}(x) \Rightarrow \text{Trace}(x)$
- $\forall x \text{NbrAttempts}(x) \Rightarrow \text{Trace}(x)$
- $\forall x \text{Course}(x) \Rightarrow \text{Situation}(x)$
- $\forall x \text{Exercise}(x) \Rightarrow \text{Situation}(x)$

- $\forall x \text{Quiz}(x) \Rightarrow \text{Situation}(x)$
- $\forall x \text{Test}(x) \Rightarrow \text{Situation}(x)$
- $\forall x \text{DW}(x) \Rightarrow \text{Situation}(x)$
- $\forall x \text{PW}(x) \Rightarrow \text{Situation}(x)$
- $\forall x \forall y \text{Situation}(x) \wedge \text{Learner}(y) \Rightarrow \exists z \text{Mark}(z) \wedge \text{hasMark}(x, y, z)$
- $\forall x \text{Situation}(x) \Rightarrow \exists y \text{Resource}(y) \wedge \text{useResource}(x, y)$

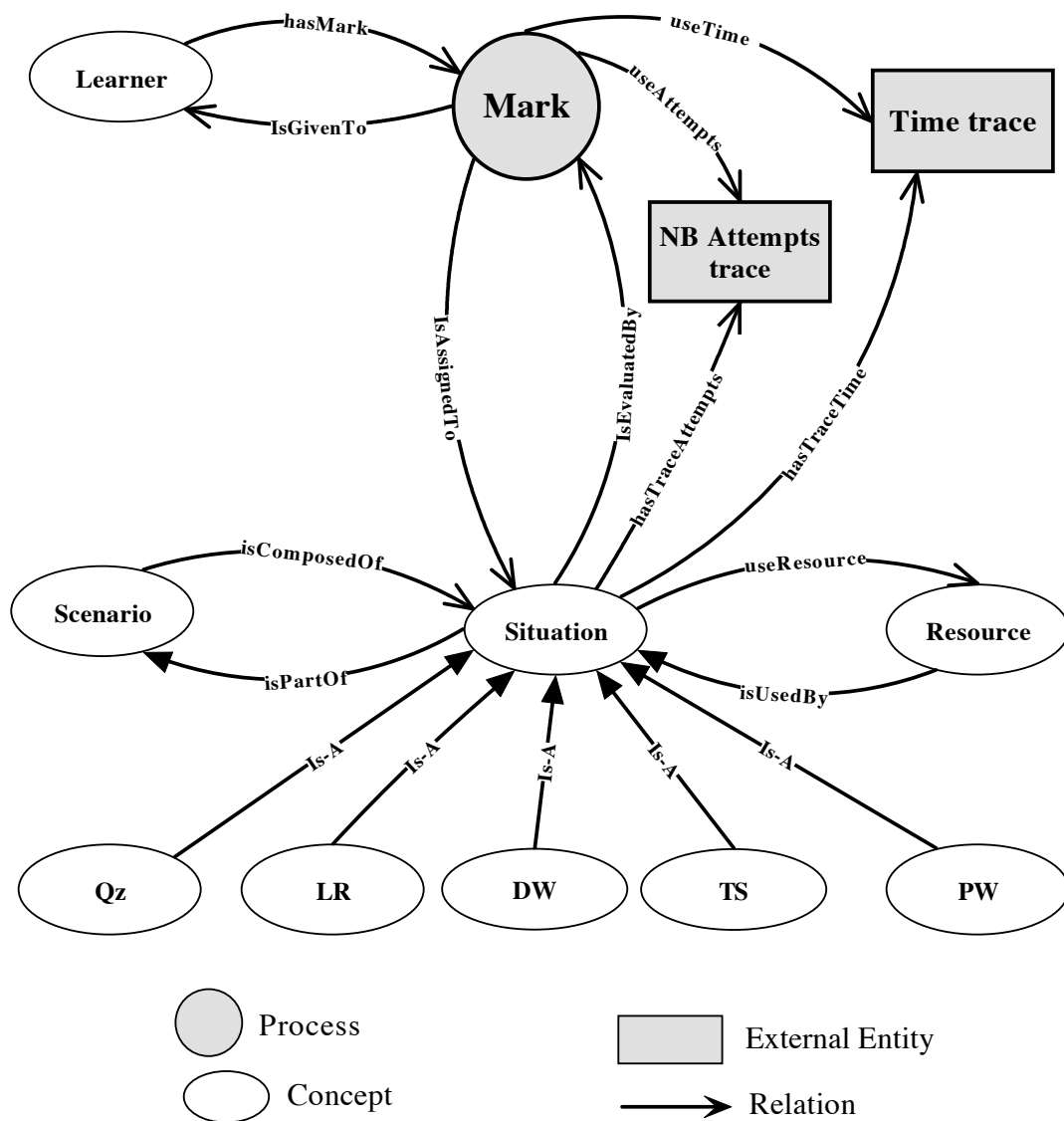


FIGURE 4.2 – Fragment de l'ontologie d'adaptation du domaine de formation.

## 4.4 Adaptation de la stratégie d'apprentissage à base de traces

Dans cette section, nous utilisons la définition d'un scénario composé de situations (Définition 3.2.5) pour construire un diagramme d'état-transition tel que présenté dans la Figure 4.3. Le diagramme illustre la séquence de situations  $S_i$  dans un scénario  $\mathcal{S}$ . Un état représente une situation  $S_i$  d'apprentissage, qui utilise un ensemble de ressources  $R = \{R_1, \dots, R_m\}$  pour atteindre un objectif  $G_i$ . Une transition entre états nécessite un ensemble de conditions à satisfaire. Les conditions sont des valeurs basées sur des traces telles qu'une note  $M_{R_j}^{S_i}$ , qui est calculée en fonction de la trace de durée ( $D_{R_j}^{S_i}$ ) et du nombre de tentatives ( $T_{R_j}^{S_i}$ ). Le processus de raisonnement dans l'ontologie est décrit dans l'algorithme 1 qui assure deux types d'adaptation :

- *Adaptation de la situation.*
- *Adaptation des ressources dans une situation.*

Pendant une activité de formation, l'apprenant interagit avec les situations en utilisant des ressources. Nous considérons les traces obtenues dans ces situations d'apprentissage. L'apprenant part d'une situation  $S_i$ ,  $i \in \{1, \dots, n\}$  et une ressource  $R_j$  de  $S_i$  où  $j \in \{1, \dots, m_i\}$  et  $m_i$  est le nombre de ressources dans la situation  $S_i$ . Le point de départ peut être défini par le système en fonction du niveau de compétence de l'apprenant. Ensuite, l'algorithme calcule une note basée sur des traces pour évaluer l'acquisition de connaissances d'apprentissage et détermine la ressource et la situation suivante comme suit :

- si  $C_1 : (M = M_{R_j})$  alors l'apprenant a atteint l'objectif dans la situation actuelle ; il peut aller à la prochaine situation avec une ressource  $R_{j+1}$  qui a un niveau de difficulté plus élevé par rapport à la précédente ;
- si  $C_2 : (M < M_{R_j}) \wedge ((D \leq D_{R_j} \wedge T > T_{R_j}) \vee (D > D_{R_j} \wedge T \leq T_{R_j}))$  alors l'apprenant n'a pas atteint l'objectif actuel ; il doit rester dans la même situation d'apprentissage mais avec une ressource  $R_{j-1}$ , c'est-à-dire avec un niveau de difficulté inférieur à celui de la ressource précédente :
- si  $C_3 : (M < M_{R_j}) \wedge (D > D_{R_j}) \wedge (T > T_{R_j})$  alors l'apprenant n'a pas atteint le but actuel ; il doit revenir à la situation précédente avec un niveau de difficulté inférieur de la ressource  $R_1$ .

Des exécutions multiples de l'algorithme 1 génère un diagramme d'état-transition comme montré sur la Figure 4.3.

---

**Algorithme 1** : Processus d'adaptation à base de traces

---

**Entrées :**

- Situation actuelle  $S_i$  et ressource  $R_j$  et leurs traces  $D_{R_j}, T_{R_j}$ .

**Sortie :** Une prochaine situation adaptée  $S_k$  ou ressource  $R_m$ .

```

1:  $M = M_{A_i}(M_{R_j}, D_{R_j}, T_{R_j})$  { $M_{R_j}$  : paramètre de la note  $R_j$ }
2: si  $M = M_{R_j}$  and  $i < n$  and  $j < m_i$  alors
3:    $S_i \leftarrow S_{i+1}$ 
4:    $R_j \leftarrow R_{j+1}^{i+1}$  { $R_{j+1}^{i+1}$  : ( $j+1$ )th resource of situation  $S_{i+1}$ }
5: sinon
6:   si  $M = M_{R_j}$  and  $i = n$  alors
7:      $level \leftarrow level + 1$ 
8:   fin si
9: sinon
10:  si ( $D \leq D_{R_j}$  and  $T > T_{R_j}$ ) or ( $D > D_{R_j}$  and  $T \leq T_{R_j}$ ) alors
11:     $S_i \leftarrow S_i$ 
12:    si  $j > 0$  alors
13:       $R_j \leftarrow R_{j-1}$ 
14:    sinon
15:      si  $i > 0$  alors
16:         $S_i \leftarrow S_{i-1}$ 
17:         $R_j \leftarrow R_j^{i-1}$ 
18:      fin si
19:    fin si
20:  fin si
21:  si  $D > D_{R_j}$  and  $T > T_{R_j}$  alors
22:    si  $i > 0$  alors
23:       $S_i \leftarrow S_{i-1}$ 
24:       $R_j \leftarrow R_1^{i-1}$ 
25:    sinon
26:       $level \leftarrow level - 1$ 
27:    fin si
28:  fin si
29: fin si

```

---



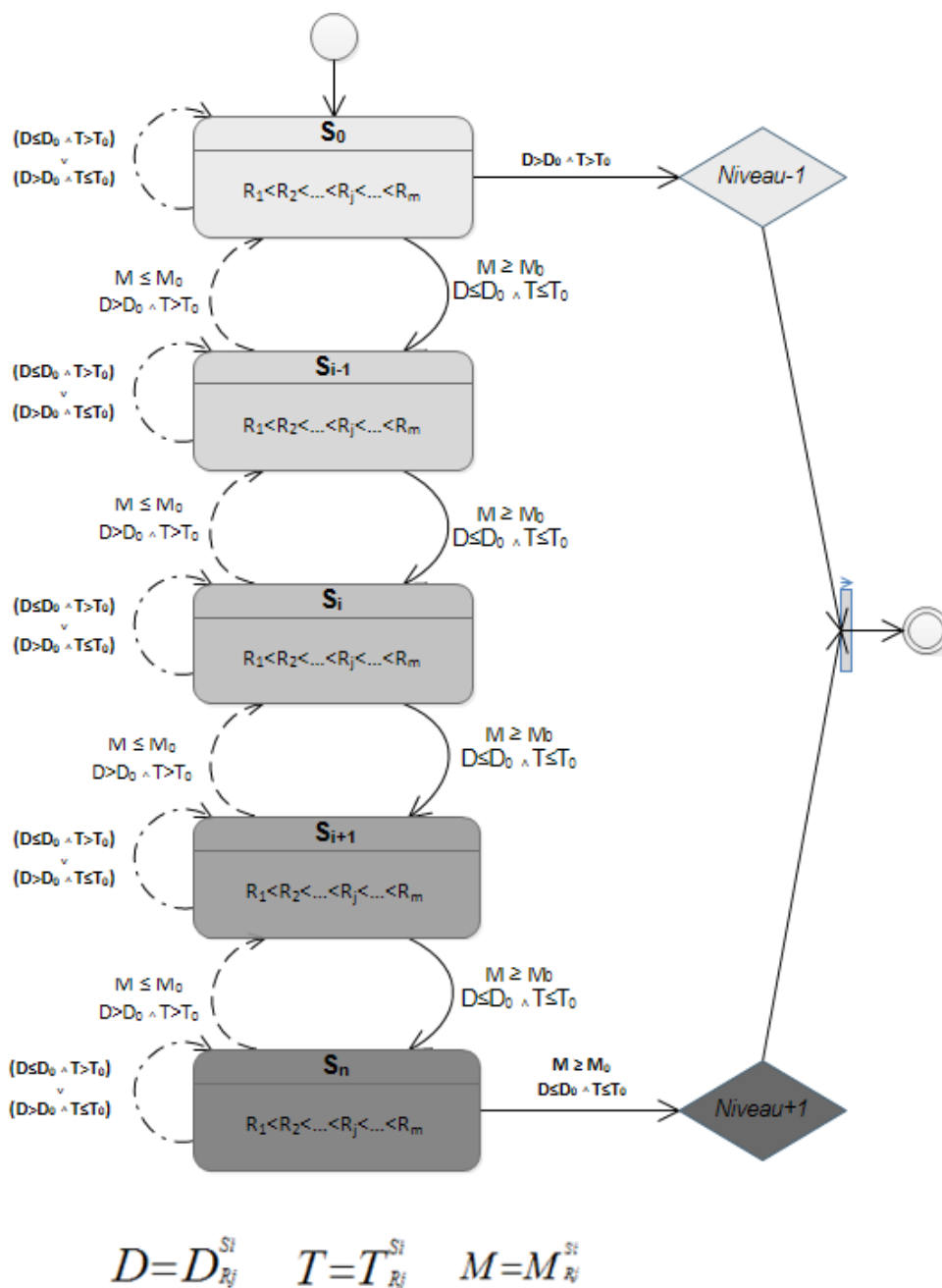


FIGURE 4.3 – Diagramme État-Transition pour l'adaptation des situations et des ressources.

**Exemple 4.4.1.** Supposons une activité d'apprentissage  $A_i$  pour la programmation en langage SQL. Supposons un scénario planifié  $\mathcal{S}$  qui se compose de trois situations  $S_i$  :

1. Cours (CR);
2. Travaux Dirigé (TD);
3. Travaux pratiques (TP).

Un scénario a un objectif global à atteindre  $G = \{\text{Select } x \text{ From } y \text{ Where } c\}$ , ce qui signifie qu'un apprenant doit acquérir la notion de requête SQL impliquant une projection et une sélection.

tion basées sur la clause WHERE. Le formateur ou l'expert fournit un ensemble de ressources  $R_{S_i}$  (qui peuvent être sous forme de texte, diapositives, vidéo ou audio) pour chaque situation  $S_i$  afin d'atteindre un objectif local  $G_{S_i}$ . Comme résumé dans la table 4.4.1, qui illustre un scénario composé de 3 situations avec leurs ressources attachées et objectifs.

Par souci de simplicité, l'enseignant fixe les mêmes paramètres pour chaque ressource :

- $M_{R_j} = 5$  qui est une note maximale complète caractérisant chaque ressource  $R_j$  dans une situation  $S_i$ , avec  $j = \{1, \dots, m_i\}$  et  $i = \{1, \dots, 3\}$ ;
- $D_{R_j} = 300$  secondes qui est la durée optimale pour compléter  $R_j$ ;
- le nombre d'essais autorisés est  $T_{R_j} = 3$  et (iv)  $\gamma = 0.2$ .

TABLE 4.4.1 – Scénario de situations avec leurs ressources

Situation	Objectif local	Ressource
Cours	Apprendre les principes théoriques de la projection et de la sélection du langage SQL	$R_1 = \text{text}$
		$R_2 = \text{slides}$
		$R_3 = \text{Video}$
Travail Dirigé (TD)	Exercer l'apprenant sur les concepts syntaxiques de la projection et de la sélection du langage SQL	$R_1 = \text{text}$
		$R_2 = \text{text}$
		$R_3 = \text{text}$
		$R_4 = \text{text}$
Travail Pratique (TP)	Pour pouvoir écrire correctement des requêtes de projection et de sélection du SQL	$R_1 = \text{video}$
		$R_2 = \text{text}$

Basé sur la Formule (3.3) nous avons obtenu les résultats résumés dans la Table 4.4.2. Nous exécutons l'algorithme 1 dans chaque situation après que l'apprenant ait terminé son interaction avec une ressource. Le résultat de l'algorithme 1 est un chemin adapté des situations et de leurs ressources à suivre pour atteindre l'objectif global.

La Figure 4.4 illustre trois scénarios selon le point de départ de l'apprenant, chacun est constitué de chemins adaptés pour différentes situations et de leurs ressources attachées. Ces chemins sont décrits comme suit :

- *chemin 1*  $p_1 = \{CR : R_1, TD : R_2, TD : R_1\}$ , l'apprenant part de la situation CR et de la ressource  $R_1$  ; il réussit la situation et passe à la situation suivante TD avec la ressource  $R_2$  ; il échoue, alors il reste dans la même situation DW avec une ressource  $R_1$  (ayant un niveau de difficulté inférieur) mais il échoue de nouveau. Dans ce cas, il n'a pas acquis les connaissances du niveau et doit revenir à Level - 1 qui est un niveau inférieur ;

- *chemin 2*  $p_2 = \{CR : R_2, TD : R_3, TP : R_2\}$ , l'apprenant commence par la situation CR avec la ressource  $R_2$ ; il atteint l'objectif et satisfait la première condition. Par conséquent, il passe à la situation suivante TD avec la ressource  $R_3$ ; L'apprenant passe la situation jusqu'à TP puis il est autorisé d'aller à un niveau supérieur  $Level + 1$ ;
- *chemin 3*  $p_3 = \{CR : R_3, TD : R_4, TP : R_2\}$ , l'apprenant débute avec la situation CR avec la ressource  $R_3$ ; il passe toutes les situations avec succès jusqu'à la situation TP. Par conséquent, il est autorisé à passer à un niveau supérieur  $Level + 1$ .

TABLE 4.4.2 – Les traces d'apprenant obtenues lors de l'exécution des situations

Situation	Ressource	D	T	M
Cours	$R_1 = text$	300	3	5.00
	$R_2 = slides$	200	1	5.00
	$R_3 = Video$	250	2	5.00
Travail Dirigé (TD)	$R_1 = text$	620	2	3.31
	$R_2 = text$	700	3	2.93
	$R_3 = text$	230	2	5.00
	$R_4 = text$	300	2	5.00
Travail Pratique (TP)	$R_1 = video$	200	1	5.00
	$R_2 = text$	300	3	5.00

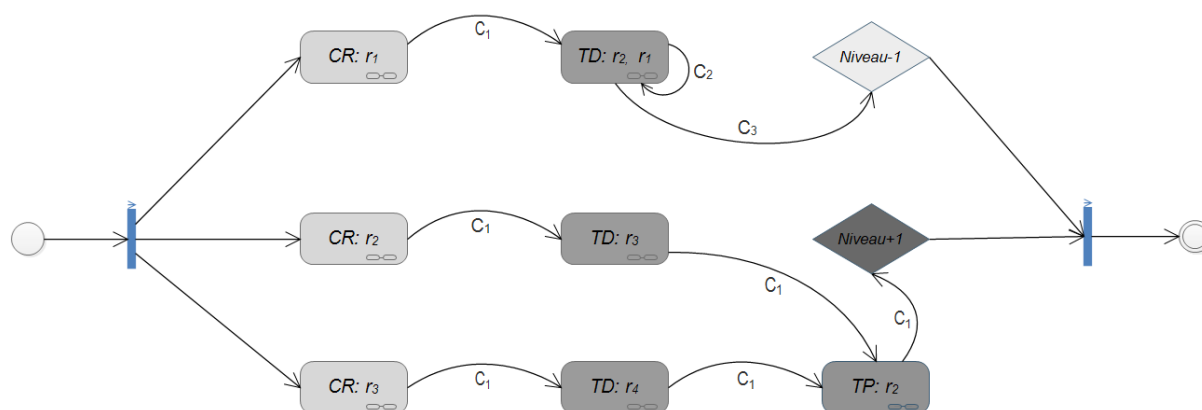


FIGURE 4.4 – Résultat de l'adaptation des situations et des ressources à base de traces.

## 4.5 Discussion et conclusion

Dans ce chapitre, nous avons présenté notre modèle d'adaptation basé sur les traces pour les ressources et la stratégie de formation. Le modèle d'adaptation est basé sur le modèle d'évaluation proposé dans le chapitre 3 qui effectue l'évaluation de l'apprenant en fonction des traces produites pendant les activités d'apprentissage [Chachoua *et al.*, 2016d].

Notre modèle d'adaptation génère automatiquement des chemins adaptés en utilisant un modèle d'état-transition. Les *états* sont des situations qui consomment des ressources. Les *transitions* sont des étapes permettant de passer d'une situation à une autre en s'appuyant sur des règles logiques et un mécanisme de raisonnement. Les deux sont détaillés dans une ontologie d'adaptation du domaine de formation et un algorithme que nous avons proposés. Le modèle d'adaptation ontologique se concentre sur cinq modèles principaux, à savoir : modèle de scénario, modèle de situation, modèle d'apprenant, modèle de ressource et modèle de trace.

Cependant, notre approche a des limitations en ce qui concerne les points suivants :

- la méthode d'adaptation nécessite la construction et la maintien de la base de connaissances ontologiques qui doit être validée par les experts du domaine ;
- Deux types de complexité sont à prendre en considération : la complexité temporelle et celle de données. Dans notre cas, la complexité temporelle est  $O(1)$  car l'algorithme proposé ne contient aucune structure de contrôle complexe (boucles, calcul de factorielles, etc.). La complexité de données, en revanche, est importante car elle est liée à la quantité de données que l'ontologie doit utiliser. Toutefois, notre ontologie de domaine est décidable car elle est similaire à une logique de description. Du point de vue pratique, la complexité de données a été résolue à l'aide du SGBD sémantique de Oracle 11g. Il assure un passage à l'échelle qui permet de gérer et de manipuler des trillions de tuples RDF <sup>1</sup>.

La prochaine étape de notre travail est la mise en œuvre et la validation de notre modèle d'adaptation basé sur les traces.

Pour ce faire, il est important d'avoir en main une collection de traces d'apprentissage modélisées. Pour cette raison, nous avons effectué une expérimentation terrain pour obtenir des données réelles.

---

<sup>1</sup><https://www.oracle.com/technetwork/database/oracle-33.pdf>

## **Partie IV**

# **Expérimentation et Évaluation**



# Table des matières

---

<b>Chapitre 5 Mise en œuvre de l’adaptation à base de traces par un modèle ontologique</b>	<b>110</b>
5.1 Introduction . . . . .	110
5.2 Approche de mise en œuvre de l’ontologie . . . . .	110
5.3 Règles d’inférences . . . . .	115
5.4 Conclusion . . . . .	126
<b>Chapitre 6 Évaluation et validation</b>	<b>128</b>
6.1 Introduction . . . . .	128
6.2 Environnement logiciel de tests . . . . .	129
6.3 Expérimentation (1) : Apprentissage avec une scénarisation classique . . . . .	129
6.4 Expérimentation (2) : Apprentissage avec une scénarisation adaptée . . . . .	134
6.5 Bilan . . . . .	145
6.6 Conclusion . . . . .	146

---

# Chapitre 5

## Mise en œuvre de l'adaptation à base de traces par un modèle ontologique

### 5.1 Introduction

Dans ce chapitre nous implémentons notre modèle d'évaluation et d'adaptation comme preuve de concept décrit dans le chapitre 4. Notre choix de modélisation est porté sur l'utilisation d'une ontologie pour pouvoir appliquer des règles de raisonnement ontologique sur des triplets RDF. Pour ce faire, nous nous sommes concentrés sur l'aspect technique de la preuve de concepts de notre modèle d'adaptation de la ressource et/ou de la situation d'apprentissage par un modèle ontologique sous « Oracle Semantic Data Store » qui nous permet un raisonnement logique sur les données ontologiques. Par conséquent, nous détaillons l'approche de mise en œuvre de l'ontologie dans la section 5.2. Nous avons détaillé dans la section 5.3, les règles d'inférence natives au SGBD Oracle, et celles liées à l'adaptation de la ressource et/ou de la situation d'apprentissage à base de traces et l'interrogation des données ontologiques. Enfin, la section 5.4 conclut le chapitre.

### 5.2 Approche de mise en œuvre de l'ontologie

#### 5.2.1 Ontologies dans les systèmes d'apprentissage

Avec la diversité des types d'apprenants, les systèmes d'apprentissage en ligne cherchent à offrir un contenu d'apprentissage individualisé et personnalisé à chaque type d'apprenant et selon ses connaissances et ses compétences. Cela exige que le système connaisse son profils, suive ses progrès pour lui offrir une formation qui répond à ses besoins. Deux types de données à collecter sur l'apprenant et qui constituent la base du modèle de personnalisation :



- données statiques : qui est une agrégation des caractéristiques des apprenants qui ne sont pas modifiées pendant un processus d'apprentissage comme, le nom, qualifications, etc. ;
- données dynamiques : qui rassemblent des informations sur la performance, les connaissances et les compétences actuelles de l'apprenant dans le contexte actuel.

En se basant sur ces deux types de données, les systèmes d'apprentissage en ligne ont adopté plusieurs approches pour la personnalisation de l'apprentissage à l'apprenant. Dans notre cas, nous nous focalisons sur les approches ontologiques. Ces dernières peuvent assurer la personnalisation du contenu pédagogique, la génération des rétroactions et la recherche de ressources d'apprentissage comme détaillé dans ce qui suit.

1. **Utilisation des ontologies pour la personnalisation du contenu** : un modèle en ligne est utilisé pour surveiller l'interaction entre l'apprenant et le système et tente d'adapter le contenu en fonction du modèle dynamique de l'apprenant Gomes *et al.* [2008]. Les concepts de cours sont organisés dans une ontologie qui représente les connaissances du domaine et la formalisation de ses concepts. La structure taxonomique est construite à partir de classes qui représentent les concepts et les relations de généralisation entre celles-ci. Avec certains modules de cours, objectifs et interactions, il existe des concepts associés. Le système peut suivre leur statut pendant la session d'apprentissage afin de déterminer les connaissances de l'apprenant. Cela permet de calculer les progrès d'apprentissage de l'apprenant. Chaque concept de l'ontologie a des paramètres associés et un état associé. Les paramètres définissent le nombre de réponses correctes et erronées ainsi que le nombre de SCO (Sharable Content Objects) achevés par l'apprenant. L'état est basé sur ces paramètres, et le système stocke des données et la manière dont un concept a été appris. Ces données sont mises à jour dans l'ontologie du domaine, ce qui permet de suivre les connaissances et le progrès de l'apprenant à tout moment ;
2. **Génération de rétroactions basée sur l'ontologie** : La rétroaction peut être décrite comme un processus ayant pour objectif d'informer l'apprenant non seulement sur l'exactitude de ses réponses, mais également sur la précision, l'actualité, les consignes d'apprentissage, les messages de motivation, les documents de base, l'objectif de l'apprentissage, etc [Mory, 2004]. Dans la génération actuelle des systèmes d'apprentissage en ligne, la rétroaction est souvent réduite à la communication par e-mail avec un tuteur, ce qui prend beaucoup de temps et inefficace. Pour remédier à cela, un mécanisme de rétroaction a été développé dans lequel les ontologies sont utilisées comme arguments du moteur de rétroaction permettant aux auteurs de définir des réactions spécifiques à un domaine ou une tâche [Passier et Jeurig, 2004]. Pour produire un retour riche en sémantique, le système doit contenir plusieurs types de connaissances utilisant les ontologies. Ces connaissances sont les suivantes :

- **Domaine** : par exemple, la technologie de communication ou programmation distribuée ;
- **Langage de modélisation** : par exemple, UML / OCL ;
- **Tâche/méthode** : par exemple, la séquence de phases au cours de laquelle un artefact est construit ;
- **Éducation** : par exemple, apprentissage de concepts, résolution de problèmes, exemples et définitions ;
- **Feedback** : par exemple, différents types de rétroaction et modèles/phases au cours des dialogues. La figure 5.1 présente l'architecture d'un système d'apprentissage en ligne prenant en charge un mécanisme de rétroaction générique.

L'architecture comprend trois composants principaux :

- **Player** : un joueur où l'apprenant peut apprendre des concepts, construire des artefacts et résoudre des problèmes ;
- **Author tool** : un outil de création permettant à un auteur de développer des cours et des contenus liés au cours (c.-à-d. des ontologies, des exemples, des modèles de feedback) ;
- **Feedback engine** : un moteur de rétroaction qui utilise des ontologies comme paramètres et génère automatiquement des commentaires. Il peut être générique, c'est-à-dire indépendant des ontologies (applicable à toutes les activités de conception) ou spécifique, défini par l'auteur et spécifique à une tâche, un domaine ou un cours.

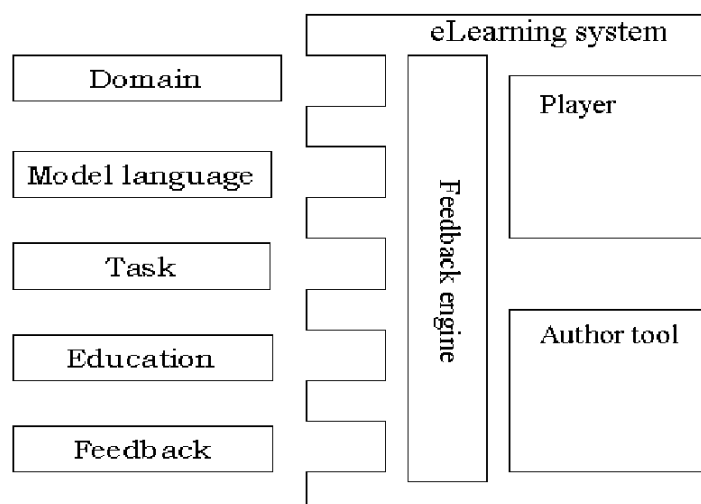


FIGURE 5.1 – Architecture fonctionnelle du mécanisme de rétroaction.

3. **Ontologie pour améliorer la recherche de ressources d'apprentissage** : Le problème commun au développement d'environnements d'apprentissage en ligne est le manque

d'interopérabilité entre les systèmes basés sur différentes ontologies. Dans [Greenberg et Robertson, 2002], les recherches ont montré que les auteurs ont du mal à annoter leurs contenus d'apprentissage en utilisant des taxonomies par sujet. Les ressources ne sont pas interconnectées sur le web, il n'y a donc pas de contexte dans lequel elles peuvent être définies. L'objectif principal est de permettre aux apprenants de trouver les ressources les plus pertinentes dans leur contexte d'apprentissage actuel. Dans [Gašević et Hatala, 2006], une structure de cours similaire à la norme SCORM a été proposée, où les ressources d'apprentissage sont liées aux structures ontologiques et annotées avec l'ontologie du cours. Comme illustré sur la figure 5.2, la partie gauche représente la structure du cours et la partie droite représente le contenu approprié. La partie inférieure contient un mécanisme de recherche permettant une recherche contextuelle de la bibliothèque numérique ACM [ACM, 2005]. Cette solution se distingue des moteurs de recherche de texte traditionnels par le fait que la demande de recherche est collectée avec l'annotation de la page en cours dans l'ontologie du cours. Ensuite, il est envoyé au système de mappage d'ontologies, qui mappe l'ontologie du cours à l'ontologie ACM CCS (Association for Computing Machinery-Computing Classification System) et envoie une requête étendue à ACM DL (Association for Computing Machinery-Digital Library). Dans [Miles et Brickley, 2005], un système SKOS (Simple Knowledge Organization) est développé qui permet de définir différents types d'ontologies (taxonomie, thésaurus) et de mapper des concepts à partir d'ontologies de domaine différentes, à l'aide de vocabulaires RDF. Cela facilite le classement ultérieur des résultats de recherche. Dans [Gašević et Hatala, 2006], les mappages définis ont permis à l'utilisateur d'améliorer la recherche dans la liste ACM DL distante.

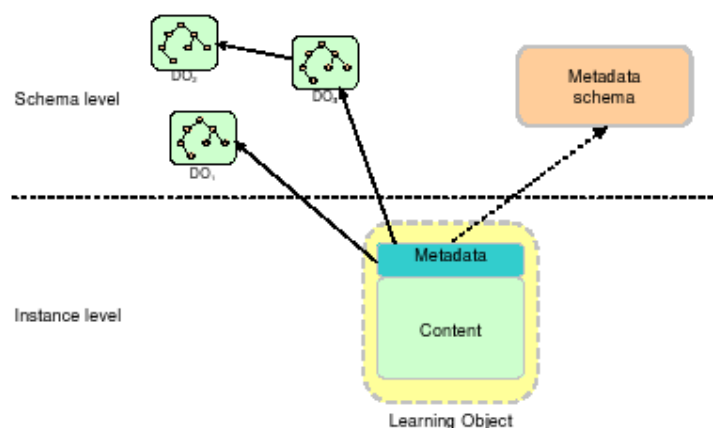


FIGURE 5.2 – Niveaux de l'ontologie d'objet d'apprentissage et de domaine [Passier et Jeuring, 2004].

Dans notre cas, nous utilisons les ontologies dans le domaine de formation en ligne afin d'adapter les ressources et/ou les situations d'apprentissage aux apprenants.

### 5.2.2 Gérer les données ontologiques

Nous nous sommes basés sur une étude comparative des bases de données RDF ou sémantique pour confirmer notre choix BDBO [Mefteh, 2013]. Le résumé de l'étude a permis de consolider notre choix qui a porté sur Oracle. Oracle peut créer des modèles de données ontologiques dans un réseau sémantique. Les données sont représentées sous forme de triplets RDF dans un modèle de graphe sémantique. Ce dernier facilite la création de plusieurs graphes séparés de modèles ontologiques. Oracle supporte les bases de règles RDFS, OWL-Prime et OWLSIF qui sont nécessaires pour les inférences. Cependant, on peut utiliser des moteurs d'inférences extérieurs comme Apache Jena<sup>1</sup> ou OpenRDF Sesame<sup>2</sup>. Enfin, le chargement des données est assuré par PLSQL et leur interrogation peut se faire avec SQL, SPARQL et ses extensions.

Nous avons exploité les technologies offertes par « Oracle Semantic Data Store » pour mettre en œuvre notre modèle d'adaptation, et ce, en créant les parties déclaratives et impératives de notre ontologie d'adaptation du domaine de formation. Après vérification de la cohérence de ce dernier, nous avons créé les règles nécessaires au processus d'adaptation de la ressource et/ou de la situation d'apprentissage. L'application de ces règles permettent d'effectuer les inférences nécessaire au processus d'adaptation.

### 5.2.3 Implantation de l'ontologie d'adaptation du domaine de formation

Pour l'implantation de l'ontologie de description du domaine de formation, nous utilisons le support « Oracle Semantic Data Store » basé sur le système de gestion de bases de données Oracle sémantique 11g. Ce dernier utilise les standards RDF (Resource Description Framework), RDFS (Resource Description Framework Schema) et le langage OWL (Web Ontology Language) pour assurer la persistance, l'inférence et l'interrogation dans une base de données à base ontologique d'Oracle. Le SGBD Oracle sauvegarde dans un schéma appelé « MDSYS » l'ontologie avec les données sous forme de triplets RDF. La mise en œuvre de notre approche d'adaptation de la ressource et/ou de la situation d'apprentissage comprend la création des parties déclaratives et impératives de l'ontologie.

### 5.2.4 Processus de validation de l'ontologie d'adaptation du domaine de formation

Il est nécessaire de vérifier la cohérence au sein de notre modèle ontologique pour s'assurer de l'absence des erreurs dans l'ontologie. Le SGBD Oracle nous donne la possibilité de vérifier

---

<sup>1</sup><https://jena.apache.org/>

<sup>2</sup><https://bitbucket.org/openrdf/sesame>

et de valider les conditions d'intégrité, exprimées dans nos modèles ontologiques, par la procédure SEM\_API.VALIDATE\_MODEL. Cette dernière permet de détecter les incohérences suivantes :

- les classes insatisfaisantes et leurs instances ;
- les triplets dont le sujet n'est pas une URI (Uniform Resource Identifier) ou un nœud vide ;
- les triplets dont le prédicat n'est pas une URI ;
- les individus qui sont différents d'eux-mêmes et qui appartiennent simultanément à deux classes disjointes ;
- les paires d'individus qui sont identiques (owl : sameAs) et différents (owl : different-From) en même temps.

### 5.3 Règles d'inférences

**Définition 5.3.1.** L'inférence est la capacité de faire des déductions logiques basées sur des règles. Elle permet la création de requêtes d'interrogation en effectuant une correspondance sémantique basée sur des relations significatives entre des éléments de données [Chuck *et al.*, 2013].

L'inférence implique l'utilisation de règles, fournies par Oracle et d'autres définies par l'utilisateur, placées dans des bases de règles.

La figure 5.3 montre les ensembles de triplets déduits du modèle de données et l'application des règles dans une ou plusieurs bases de règles. Dans cette illustration, la base de données peut avoir un nombre quelconque de modèles sémantiques, de bases de règles et d'ensembles de triplets déduits. Un ensemble de triplets déduits peut être dérivé en utilisant des règles d'une ou de plusieurs bases de règles.

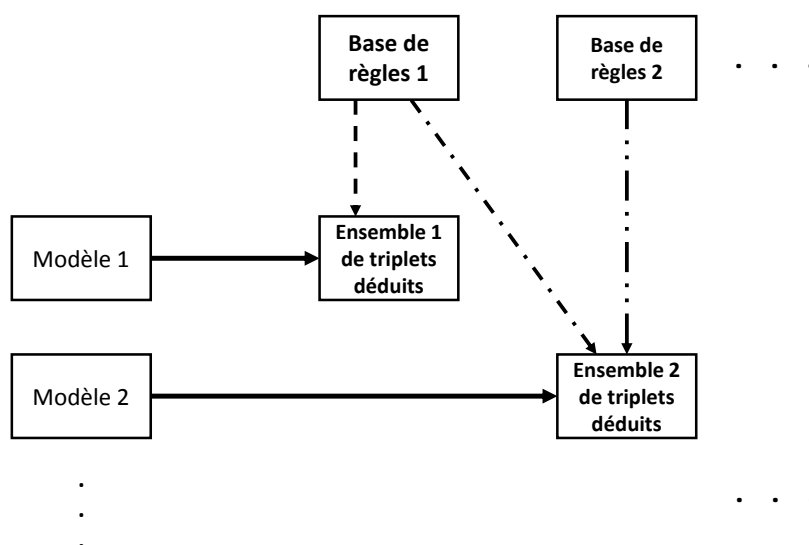


FIGURE 5.3 – Inférence à base de règles dans un modèle ontologique [Chuck *et al.*, 2013]

**Définition 5.3.2.** Une *règle* est un objet qui peut être appliqué pour obtenir des inférences à partir de données sémantiques [Chuck *et al.*, 2013]. Une règle est identifiée par un nom et comprend :

- la partie IF : un modèle pour les antécédents ;
- la partie filtre (conditions) : qui restreint davantage les sous-graphes correspondant au motif de la partie IF ;
- la partie latérale THEN : les conséquences.

Par conséquent, nous pouvons interroger les données sémantiques déduites par l'inférence à base de règles dans nos modèles ontologiques.

### 5.3.1 Base de règles d'inférence d'Oracle

Oracle Database prend en charge la base de règles pour les vocabulaires RDFS et une partie de OWL, qui ont une expressivité importante, ainsi que OWL 2 RL. Chaque vocabulaire pris en charge a une base de règles correspondante ; ces bases de règles n'ont pas besoin d'être peuplées car les règles d'implication sous-jacentes aux vocabulaires sont implémentées en interne [Chuck *et al.*, 2013]. Les bases de règles fournies par Oracle sont les suivants :

1. RDF : présentation des triplets dans l'ontologie (sujet, prédicat, objet) ;

2. RDFS : extension du RDF par les constructeurs définis dans la section 5.2 ;
3. OWLSIF : OWL avec la sémantique de IF « IF Semantic », le vocabulaire et la sémantique proposés pour RDF ;
4. RDFS++ : une extension minimale à RDFS qui est RDFS plus « owl :sameAs et owl :InverseFunctionalProperty » ;
5. OWLPrime : Un sous-ensemble du langage OWL défini par Oracle comme éléments du standard OWL pris en charge par le moteur d'inférence natif « Oracle Database semantic technologies ». Les capacités d'OWL sont :
  - Notions de base : classe, sous-classe, propriété, sous-propriété, domaine, type ;
  - Caractéristiques de la propriété : transitive, symétrique, fonctionnelle, inverse fonctionnelle, inverse ;
  - Comparaisons de classes : équivalence, disjonction ;
  - Comparaisons de propriétés : équivalence ;
  - Comparaisons individuelles : identiques, différentes ;
  - Expressions de classe : complément ;
  - Restrictions de propriété : hasValue, someValuesFrom, allValuesFrom.

### 5.3.2 Règles d'inférence métiers

Nous avons défini des règles d'inférence métiers liées à notre domaine d'application qui est l'adaptation. Ces règles sont les suivantes :

- La règle « LearnerFailedSituation » à base de la note obtenue : permet d'obtenir la liste des apprenants qui ont échoué dans une situation d'apprentissage ;
- La règle « learnerSuccesGoNextSituation » à base de la note obtenue : permet d'obtenir la liste des apprenants qui ont réussi dans une situation d'apprentissage.

Ces règles nous ont permis de mettre en œuvre notre processus d'adaptation de la ressource et/ou de la situation pédagogique à l'apprenant en se basant sur l'évaluation fondée sur les traces produites durant une activité d'apprentissage. Les règles créées sont sauvegardées dans la base de règles « learnerFailedSituation\_rb ». Ci-dessous, nous détaillons la création de la règle liée à l'échec des apprenants aux situations d'apprentissage en se basant sur leurs notes.

```

1. EXECUTE SEM_APIS.CREATE_RULEBASE('learnerFailedSituation_rb');
2.
3. INSERT INTO mdsys.semr_learnerFailedSituation_rb
4. VALUES('failing_rule',
5. '(?lObj1 rdf:type os:Learner)(?sObj2 rdf:type os:Situation)(?sObj2 os:hasFixedMark ?fMark)(?lObj1 os:
   doSituation ?sObj2)(?lObj os:hasMark ?mark)',
6. '(fMark = 5)and (mark < 5)',
7. '(?lObj1 os:LearnerFailedSituation ?sObj2)',
8. SEM_ALIASES(SEM_ALIAS('os','http://l3i.univ-lr.fr/eadaptOnto')));

```

### Code 5.3.1 – Création de la règle échec d'apprenant

Dans le code 5.3.1, la ligne (1) crée la règle « learnerFailedSituation ». Cela permet dans la ligne (3) la création d'une vue nommée « semr\_learnerFailedSituation » dans le schéma d'Oracle mdsys. Ligne (4) représente le nom unique pour chaque règle insérée dans la base. La partie condition IF est représentée sous forme de triplets RDF dans la ligne (5). Le filtre dans la ligne (7) permet de limiter le sous-graphe défini par la partie conditionnelle. Ligne (8) représente la conséquence de l'application de la règle. Enfin, la ligne (9) précise l'espace de noms utilisé pour la règle créée. Dans cette règle, le paramètre (fMark = 5) représente la note optimale fixée par l'enseignant ou l'expert pour une situation donnée. La valeur de l'expression logique (mark < 5) représente la note obtenue par l'apprenant et qui est inférieure à la note optimale. Cette condition cible les apprenants qui ont échoué dans l'activité. Dans ce qui suit, les règles sont formalisées de la même manière que pour la règle « LearnerFailedSituation ».

### 5.3.3 Les éléments de l'ontologie

Dans le chapitre 4, nous avons défini trois traces sur lesquelles nous avons fondé notre modèle d'adaptation de la ressource et/ou de la situation d'apprentissage aux l'apprenants. Ces traces sont :

- la trace temporelle (durée prise par un apprenant pour répondre à un problème);
- la trace de tentatives (le nombre d'essais fait par un apprenant pour résoudre un problème);
- la combinaison des deux traces temporelle et de tentatives, la note obtenue à base des deux traces.

A cet effet, nous avons défini dans l'ontologie « eadaptOnto » les propriétés essentielles à notre modèle d'adaptation comme l'illustre le Code 5.3.2. Le détail des différents éléments de l'ontologie en termes de classes, propriétés et processus est présenté dans les tableaux 5.1, 5.2 et 5.3.



```
hasDurationTrace, hasAttemptsTrace, hasMarkTrace, useDurationTrace, useAttemptsTrace
```

### Code 5.3.2 – Relations de traces liées aux règles d'ontologie eadaptOnto

En plus, d'autres relations des différents concepts de l'ontologie sont définies dans le Code 5.3.3. Ces relations permettent la création de règles combinées pour exprimer des règles d'inférence plus complexes dans le modèle ontologique.

```
doSituation, useRessource, isRelatedTo, isComposedOf, isAssignedTo, isPartOf, etc.
```

### Code 5.3.3 – Relations supplémentaires liées aux règles d'ontologie eadaptOnto

Une règle d'inférence peut faire usage d'une ou de plusieurs relations décrites dans sa partie impérative. Dans ce qui suit, nous décrivons les règles d'inférence à base des relations en faisant usage des différentes traces susmentionnées. Les traces produites durant l'activité d'apprentissage de l'apprenant sont nécessaires à l'établissement des règles ontologiques pour adapter en conséquence la ressource et/ou la situation pédagogique. Les différentes règles, établies dans notre modèle ontologique, nous assurent l'implémentation de notre Algorithme 1 d'adaptation à base de traces décrit dans la section 4.4 du chapitre 4.

TABLE 5.1 – Modèle d'évaluation et système d'adaptation à base de traces

Modèle/Systeme	Description
Notation (évaluation)	C'est un modèle qui évalue l'activité de l'apprenant en se basant sur les traces temporelles et le nombre de tentatives.
Adaptation de la ressource et/ou la situation d'apprentissage	C'est un système qui met en œuvre le processus de raisonnement sur l'ontologie en tenant compte des traces de l'activité pour générer des situations et des ressources adaptées aux apprenants.

TABLE 5.2 – Hiérarchie de classes de l'ontologie du domaine de la formation

	<b>Classe</b>	<b>Sous-classe</b>	<b>Description</b>
Hiérarchie de classes	Situation	Lecture (LR)	c'est un cours présenté par un enseignant.
		DirectedWork (DW)	C'est le contenu d'apprentissage comme des exercices et des problèmes dédiés à approfondir les connaissances sur le sujet étudié dans un cours.
		PracticalWork (PW)	C'est un contenu d'apprentissage pour expérimenter les connaissances acquises pendant une session de cours et/ou d'un travail dirigé.
		Test (TS)	C'est un examen subi par les apprenants afin d'évaluer leur acquisition de connaissances.
		Quiz (QZ)	C'est un ensemble de questions sur les connaissances ciblées dans CR, TD et TP.
	Scenario	-	C'est une succession d'exécution de situations afin d'atteindre un objectif global. Un scénario est conçu par un expert ou un formateur.
	Resource	-	C'est un contenu d'apprentissage que les apprenants peuvent utiliser dans une situation donnée.
	Learner	-	C'est un acteur qui interagit avec les ressources dans une situation de formation.
	Trace	TimeTrace	C'est la durée mise par un apprenant pour accomplir l'objectif de la situation.
		AttemptsTrace	C'est le nombre de tentatives prises par l'apprenant pour atteindre l'objectif dans une situation.

TABLE 5.3 – Les objets de propriétés de l'ontologie du domaine de la formation

Objets de propriétés	Description
doSituation	Décrit la relation d'interaction de l'apprenant avec la situation d'apprentissage.
useRessource	Désigne les ressources utilisées dans une situation d'apprentissage.
isComposedOf	Un scénario est composé d'une ou de plusieurs situations.
isAssignedTo	Une note maximale est fixée par un formateur dans une situation d'apprentissage.
isGivenTo	Une note obtenue par l'apprenant dans une situation d'apprentissage.
is-A	LR, DW, PW, TS, QZ Se sont des sous-classes de la classe Situation.
useDurationTrace	Le calcul de la note est basé sur la trace temporelle.
useAttemptsTRace	Le calcul de la note est basé sur la trace de tentatives.
hasMarkTrace	La trace de la note obtenue par l'apprenant dans une situation d'apprentissage.
hasDurationTrace	La trace temporelle correspondant à la durée que l'apprenant a met pour effectuer une situation d'apprentissage.
hasAttemptsTrace	La trace du nombre de tentatives correspondant aux nombre d'essais que l'apprenant a effectué dans une situation d'apprentissage.

### 5.3.4 Règle d'inférence pour le processus d'adaptation

#### Adaptation de la situation basée sur la note

La règle « learnerGoadaptedSituation » décrite ci-dessous permet d'adapter la situation d'apprentissage en se basant sur la note obtenue par l'apprenant pendant son apprentissage. La règle illustrée dans le code 5.3.4 précise que si un apprenant obtient une note inférieure à la note optimale fixée par l'enseignant dans une situation donnée alors il n'a pas acquis les compétences requises pour passer à la situation suivante. Dans ce cas, il doit refaire une situation adaptée pour renforcer son apprentissage et passer à un autre niveau de compétence.

```

1. EXECUTE SEM_APIS.CREATE_RULEBASE('adaptedSituation_rb');
2.
3. INSERT INTO mdsys.semr_adaptedSituation_rb
4. VALUES('learnerGoadaptedSituation_rule',
5. '(?lObj1 rdf:type os:Learner)(?sObj3 rdf:type os:Lecture)(?sObj3 os:doneByLearner ?lObj1)(?lObj1 os:
   hasMarkTrace ?mark)',
6. '(mark <5)',
7. '(?lObj1 os:learnerGoadaptedSituation ?sObj3)',
8. SEM_ALIASES(SEM_ALIAS('os','http://13i.univ-lr.fr/eadaptOnto')));

```

Code 5.3.4 – Création de la règle learnerGoadaptedSituation\_rule

Dans le code 5.3.4, la ligne (1) permet la création de la règle « learnerGoadaptedSituation » qui produit la vue « semr\_adaptedSituation\_rb ». Cette dernière est accessible en lecture ou en écriture par son créateur en ligne (3). Cette nouvelle règle insérée dans la base de règles a un nom unique qui est indiqué par la ligne (4). Les différentes relations qui portent sur l'apprenant, la situation et la note sont illustrées par la ligne (5). Le filtrage des données d'inférence est assurée par la condition exprimée sur la notation comme le montre la ligne (6). La relation déduite dans la ligne (7) correspond aux résultats inférés. Enfin, la ligne (8) déclare l'espace de nom utilisé pour cette règle.

### Adaptation de la ressource basée sur la note

La règle « learnerGoadaptedRessource » décrite ci-dessous permet d'adapter la ressource utilisée dans une situation d'apprentissage en se basant sur la note obtenue par l'apprenant pendant son apprentissage. La règle décrite dans le code 5.3.5 précise que si un apprenant obtient une note complète dans une ressource utilisée d'une situation d'apprentissage alors il a acquis les compétences requises pour passer à la ressource suivante. Dans ce cas, il doit réussir toutes les ressources de la situation pour passer à la situation suivante.

```

1. EXECUTE SEM_APIS.CREATE_RULEBASE('adaptedRessource_rb');
2.
3. INSERT INTO mdsys.semr_adaptedRessource_rb
4. VALUES('learnerGoadaptedRessource_rule',
5. '(?lObj1 rdf:type os:Learner)(?sObj2 rdf:type os:Lecture)(?rObj3 rdf:type os:CoursRessource)(?rObj4 rdf:
   type os:TdRessource)(?sObj2 os:useResource ?rObj3)(?lObj1 os:doSituation ?sObj2)(?lObj1 os:hasMark ?mark)
   ',
6. '(mark = 5)',
7. '(?lObj1 os:learnerGoadaptedRessource ?rObj4)',
8. SEM_ALIASES(SEM_ALIAS('os','http://l3i.univ-lr.fr/eadaptOnto')));

```

### Code 5.3.5 – Création de la partie impérative de la règle learnerGoadaptedRessource\_rule

Dans le code 5.3.5, la ligne (1) permet la création de la règle « learnerGoadaptedRessource » qui produit la vue « semr\_adaptedRessource\_rb ». Cette nouvelle règle insère dans la base de règles un nom unique qui est indiqué par la ligne (4). Les différentes relations qui portent sur l'apprenant, la situation, la ressource consommée dans une situation et la note sont illustrées par la ligne (5). Le filtrage des données d'inférences est assurée par la condition exprimée sur la notation comme le montre la ligne (6). La relation déduite dans la ligne (7) est le résultat inféré. Enfin, la ligne (8) déclare l'espace de nom utilisé pour cette règle.

### Adaptation de la ressource basée sur la note, la durée et le nombre de tentatives

Dans cette sous-section, nous avons combiné trois données, à savoir : la note, la trace temporelle et la trace de tentatives pour adapter la ressource et/ou la situation d'apprentissage à

l'apprenant. La règle détaillée dans le code 5.3.6 montre que si un apprenant obtient une note inférieure à la note optimale, une durée qui dépasse la durée optimale et un nombre d'essais supérieur à celui fixé par le formateur dans une ressource consommée par une situation donnée alors il n'a pas acquis les compétences requises pour passer à la ressource suivante dans la même situation. Dans ce cas, il doit refaire et réussir la ressource précédente de la situation actuelle ou de la situation précédente pour pouvoir passer à une autre étape d'apprentissage.

```

1. EXECUTE SEM_APIS.CREATE_RULEBASE('adaptedSituationAndRessource_rb');
2.
3. INSERT INTO mdsys.semr_adaptedSituationAndRessource_rb
4. VALUES('learnerGoadaptedSituationRessource_rule',
5. '(?lObj1 rdf:type os:Learner)(?sObj2 rdf:type os:DirectedWork)(?rObj3 rdf:type os:SimpleSelectiontd)(?sObj3
   rdf:type os:Lecture)(?rObj4 rdf:type os:SimpleSelectioncours)(?sObj3 os:useResource ?rObj4)(?sObj2 os:
   useResource ?rObj3)(?lObj1 os:doSituation ?sObj2)(?lObj1 os:hasMark ?mark)(?lObj1 os:hasDuration ?duration
   )(?lObj1 os:hasAttempts ?attempts)',
6. '(mark < 5)and(duration >50)and(attempts > 3)',
7. '(?lObj1 os:learnerGoadaptedSituation ?sObj3)(?lObj1 os:learnerGoadaptedRessource ?rObj4)',
8.SEM_ALIASES(SEM_ALIAS('os','http://l3i.univ-lr.fr/eadaptOnto')));

```

Code 5.3.6 – Création de la règle learnerGoadaptedSituationRessource\_rule

Dans le code 5.3.6, la ligne (1) permet la création de la règle « learnerGoadaptedSituation » qui produit la vue « semr\_adaptedSituationAndRessource\_rb ». Cette dernière est accessible en lecture ou en écriture par son créateur en ligne 3. Cette nouvelle règle insère dans la base de règles un nom unique exprimé par la ligne (4). Les différentes relations qui portent sur l'apprenant, la situation, la ressource consommée dans une situation, la note, trace temporelle et trace de tentative sont illustrées par la ligne (5). Le filtrage de données d'inférences est assurée par la condition faite sur la notation, trace temporelle et trace de tentative comme le montre la ligne (6). La relation déduite dans la ligne (7) est le résultat. Enfin, la ligne (8) déclare l'espace de nom utilisé pour cette règle.

Nous pouvons définir des règles d'adaptation de la situation ou de la ressource qui combinent deux traces uniquement. Dans ce cas, le filtre ne porte que sur la note et la trace temporelle ou sur la note et celle du nombre de tentative. De ce fait, c'est à l'expert de procéder à la construction et à l'enrichissement de la base de règles suivant les besoins et exigences de la formation.

### 5.3.5 Interrogation de la base de données ontologique

Les langages d'interrogation peuvent être généralement classés en trois catégories : basés sur RDF, appliquant une correspondance de sous-graphe de triplets RDF au graphe ontologique ; Basé sur les DL (Description Logics) [Calvanese *et al.*, 2007], supportant directement la sémantique DL. La plupart des langages de requête utilisés pour interroger les données RDF sont SPARQL [Harris *et al.*, 2013; Prud *et al.*, 2006], SeRQL [Aduna et Sirma] et RQL

[Karvounarakis *et al.*, 2004]. Ces trois langages offrent la prise en charge de fonctionnalités telles que la transformation de graphe ou la spécification de règles simples [Shaw *et al.*, 2012]. Le langage de requête SPARQL est devenu un standard du W3C et supporté par Oracle sémantique 11g. C'est pour cela, il est recommandé pour interroger des modèles de graphe qui correspondent aux triplets RDF.

Oracle 11g prend en charge RDFS/OWL dans son moteur de raisonnement OWLPrime [Chuck *et al.*, 2013]. Son moteur d'inférence est implémenté en tant qu'application de base de données. Il prend en charge des qui étend SQL avec de nouvelles constructions pour interroger RDF dans le SGBD relationnel d'Oracle. L'inférence implique l'utilisation de règles, soit fournies par le raisonneur, soit définies par l'utilisateur.

Le moteur d'inférence d'Oracle exécute des règles d'inférence pour un modèle RDF basé sur les données sauvegardées sur une table de triple (table de données sémantique). L'inférence est un processus de dérivation de relations implicites (nouveaux triplets). Chaque triplet créé à partir de l'exécution des règles d'inférence est également stocké dans la table de données sémantique. Ce moteur d'inférence calcule l'implication basée sur des règles de production et Oracle prend en charge deux types d'implication [Das et Srinivasan, 2009], à savoir :

- Implication standard : qui prend en charge les modèles de données sémantiques RDF, RDFS et OWL. La prise en charge de RDF et de RDFS est simplifiée par la disponibilité d'axiomes et de règles représentant leur sémantique. La prise en charge des principaux sous-ensembles de vocabulaires OWL-Lite et OWL-DL a été fournie ;
- Implication personnalisée : comme les vocabulaires standard ne peuvent pas traiter toutes les variétés des données d'application sémantique, il devient important de prendre en charge l'implication basée sur des règles arbitraires définies par l'utilisateur.

A cet effet, nous utilisons le moteur d'inférence d'Oracle pour l'inférence de données sémantiques à base de règles et SQL comme un langage d'interrogation de notre modèle ontologique « eadaptOnto ».

Avant toute interrogation de la base de données ontologique, nous devons lancer le processus d'inférence. Ce dernier est pris en charge dans Oracle par la procédure « CREATE\_ENTAILMENT » du package SEM\_API qui assure la création d'un index de règles d'inférence, dans notre cas il est nommé « eadaptOnto\_idx » comme décrit dans le code 5.3.7. Ce dernier contient les données pré-calculées qui peuvent être déduites par l'utilisation d'une ou de plusieurs règles d'inférences sur l'ensemble des modèles ontologiques créés. Une vue « SEMI\_eadaptOnto\_idx » est créée dans MDSYS pour permettre l'accès aux triplets de données RDF inférées.

```
1. EXECUTE SEM_API.CREATE_ENTAILMENT(  
2. 'eadaptOnto_idx',  
3 sem_models('eadaptOnto'),  
4. sem_rulebases('OWLPRIME',  
5. 'adaptedSituation_rb',  
6. 'adaptedRessource_rb',  
7. 'adaptedSituationAndRessource_rb'),  
8. SEM_APIS.REACH_CLOSURE, null, 'USER_RULES=T');
```

#### Code 5.3.7 – Création du processus d'inférence

Dans le code 5.3.7, la ligne (1) et (2) créent un index d'inférence nommé « eadaptOnto\_idx ». La ligne (3) précise le modèle ontologique utilisé dans le processus d'inférence. Les lignes (5, 6, 7) représentent les bases de règles utilisées pour l'inférence. OWLPrime garantit la mise en correspondance de l'ontologie avec les différentes bases de règles. La ligne (8) précise les caractéristiques suivantes de l'index d'inférence :

- paramètre 1 : le nombre de tours que le moteur d'inférence doit exécuter, par défaut c'est « SEM\_APIS.REACH\_CLOSURE » ;
- paramètre 2 : représente les composants d'inférence, la valeur par défaut est NULL ;
- paramètre 3 : « USER\_RULES=T » cette option est requise lors de l'application de règles définies par l'utilisateur.

L'interrogation de la base de données ontologique dans Oracle est assurée par la procédure « SEM\_MATCH » comme détaillé dans le Code 5.3.8. La description des paramètres est comme suit :

- query : un ou plusieurs modèles de triplets ;
- models : un ou plusieurs noms de modèles à interroger ;
- rulebases : un ou plusieurs noms de base de règles créées par l'utilisateur ou natives à Oracle ;
- aliases SEM\_ALIASES : une ou plusieurs ontologies à utiliser ;
- filter : tout critère de sélection supplémentaire ;
- index\_status : index de base de règles, qui peut avoir trois statuts, à savoir, valide, complet ou incomplet ;
- graphs SEM\_GRAPHS, named\_grahps : l'interrogation peut porter sur différents graphes identifiés par leurs noms.

```

1. SEM_MATCH(
2. query VARCHAR2,
3. models SEM_MODELS,
4. rulebases SEM_RULEBASES,
5. aliases SEM_ALIASES,
6. filter VARCHAR2,
7. index_status VARCHAR2,
8. options VARCHAR2,
9. graphs SEM_GRAPHS,
10. named_graphs SEM_GRAPHS ) RETURN ANYDATASET;

```

Code 5.3.8 – Définition de la procédure SEM\_MATCH

**Exemple 5.3.1.** *Nous avons formulé une requête (5.3.10) pour interroger les données inférées en se basant sur la base de règle « adaptedSituationAndRessource\_rb » créée dans le Code 5.3.6. La requête est présentée dans le Code ci-dessous :*

*Trouver la situation et la ressource adaptée à l'apprenant qui a accompli un TD en utilisant une ressource sur la simple selection dans SQL et qu'il ne l'a pas réussie en produisant les traces suivantes: note < 5, durée > 30 minutes et nombres de tentatives > 3 essais.*

Code 5.3.9 – Définition de la procédure SEM\_MATCH

```

1. SELECT lObj1 "Learner", sObj3 "Situation", rObj4 "Ressource"
2. FROM TABLE(SEM_MATCH(
3. '(?lObj1 :learnerGoadaptedSituation ?sObj3)(?lObj1 :learnerGoadaptedRessource ?rObj4)(?lObj1 rdf:type :
   Learner)',
4. SEM_Models('eadaptOnto'),
5. SEM_Rulebases('RDFS','adapt_rb12'),
6. SEM_ALIASES(SEM_ALIAS('os','http://l3i.univ-lr.fr/eadaptOnto')),
7. null));

```

Code 5.3.10 – Formulation de la requête 5.3.9

## 5.4 Conclusion

Ce chapitre décrit la mise en œuvre de l'adaptation basée sur l'ontologie dans le SGBD Oracle. Pour ce faire, nous avons implanté l'ontologie définie pour notre processus d'adaptation en utilisant le support « Oracle Semantic Data Store » basé sur le système de gestion de bases de données Oracle sémantique 11g. Ce dernier utilise les standards RDF et RDFS qui nous ont permis la persistance, l'inférence et l'interrogation dans une base de données à base ontologique. La mise en œuvre du processus d'adaptation nécessite la création de règles d'inférences dans des base de règles. À cet effet, nous avons défini des règles d'inférences métiers liées à notre champ d'étude. Ces règles d'inférence sont les suivantes :



- La règle « LearnerFailedSituation » à base de la note obtenue : permet d'obtenir la liste des apprenants qui ont échoué dans une situation d'apprentissage.
- La règle « learnerSuccesGoNextSituation » à base de la note obtenue : permet d'obtenir la liste des apprenants qui ont réussi dans une situation d'apprentissage.

Nous avons défini d'autres règles qui assurent l'adaptation de la ressource et/ou la situation d'apprentissage telles que :

- La règle « learnerGoadaptedSituation » permet d'adapter la situation d'apprentissage en se basant sur la note obtenue par l'apprenant pendant son apprentissage.
- La règle « learnerGoadaptedRessource » permet d'adapter la ressource utilisée dans une situation d'apprentissage en se basant sur la note obtenue par l'apprenant pendant son apprentissage.
- La règle « learnerGoadaptedSituation » permet d'adapter la situation d'apprentissage en se basant sur la note, la trace temporelle et le nombre de tentatives.

Dans le chapitre 6, nous présentons une expérimentation basée sur nos modèles d'évaluation et de notre approche d'adaptation à base de traces sur des données réelles.

# Chapitre 6

## Évaluation et validation

### 6.1 Introduction

Ce chapitre présente la validation des contributions à travers des expérimentations de nos approches d'évaluation de l'apprenant et de l'adaptation de la ressource et de la situation d'apprentissage. Pour cette expérimentation, nous avons choisi une population d'étudiants en Licence Informatique (L2) en collaboration avec l'université de Bab Ezzouar-Alger.

L'expérimentation a été faite durant quatre semaines en apprentissage du langage SQL sur deux groupes de quinze étudiants chacun. Nous avons développé un scénario d'apprentissage en deux étapes. Une première étape en scénarisation classique suivie par une évaluation classique du travail de l'apprenant et une deuxième étape en scénarisation adaptée dans laquelle nous avons activé nos approches d'évaluation et d'adaptation. À partir des résultats obtenus des deux phases, nous avons constaté une nette amélioration d'acquisition en processus d'apprentissage adaptée en comparaison avec celui du processus classique.

Nous présentons dans la section 6.2 l'environnement logiciel de tests et ses composantes. La section 6.3 décrit l'expérimentation en apprentissage classique avec une scénarisation classique pour le premier groupe d'apprenants. Les résultats sont illustrés sous forme de graphes et de statistiques et analysés en prenant en considération les notes obtenues durant chaque situation d'apprentissage. La section 6.4 présente la deuxième phase de notre expérimentation. Nous considérons un autre groupe d'apprenants qui ont suivi un apprentissage adapté pour chaque situation d'apprentissage fondé sur les traces durant la session d'apprentissage. Ces traces ont servi à l'évaluation des apprenants et à l'adaptation de la situation et/ou de la ressource pédagogique. Enfin, nous terminons par un bilan qui montre les limites relevées de l'expérimentation à base de nos modèles d'évaluation et notre approche d'adaptation fondée sur les traces d'apprentissage. La section 6.6 conclut ce chapitre.

## 6.2 Environnement logiciel de tests

L'environnement de test est implémenté sous une architecture trois tiers, comme illustré sur la figure 6.1 :

1. un client web simple qui a pour rôle de permettre à un utilisateur de se connecter à son environnement d'apprentissage, de lui présenter la ressource courante dans la situation actuelle, et de collecter les traces temporelles et de nombre de tentatives ;
2. la base de données ontologique qui contient les situations et les ressources ;
3. un serveur de servlets java qui traite les traces temporelles et du nombre de tentatives, et implémente l'algorithme d'adaptation en se connectant aussi à la base de données ontologique, et répond à l'utilisateur.

Dans le cadre de cette expérimentation, nous considérons une modélisation simplifiée des traces, nous avons utilisé un modèle dans lequel une trace modélisée est une ligne de base de données avec les attributs [IDTrace, IDApprenant, Type = {Time, Attempts}, Valeur].

L'apprenant doit se connecter à l'aide d'un nom d'utilisateur et un mot de passe et il est rattaché à une situation en cours, la ressource dans la situation et les traces d'apprentissage.

Les ressources sont présentées sous forme de texte, lien vers les fichiers PowerPoint et des liens vers des fichiers vidéos dans un répertoire externe.

Le client web offre à l'apprenant un espace d'affichage et des boutons « Suivant », et « Pause/Reprendre » pour la gestion de son avancement durant l'apprentissage.

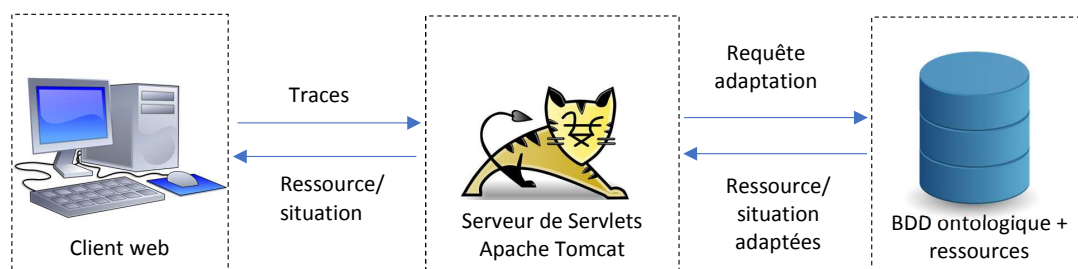


FIGURE 6.1 – Environnement logiciel de tests.

## 6.3 Expérimentation (1) : Apprentissage avec une scénarisation classique

Nous avons défini un scénario d'apprentissage pour une classe en L2 en Informatique composée de 15 étudiants. Le scénario a les objectifs suivants :

1. un objectif général à atteindre  $G = \{\text{maîtrise le langage de définition de données (LDD)}\}$  ;
2. des objectifs locaux  $G' = \{\text{création de table (CREATE), suppression de table (DROP), modification de table (ALTER)}\}$ .

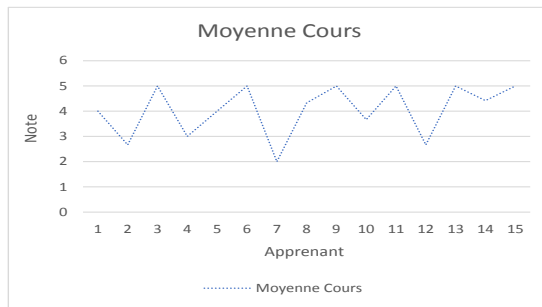
Ce qui signifie que l'apprenant doit assimiler la notion de création, de suppression et de modification de tables SQL. Le scénario est composé de quatre situations pédagogiques, à savoir : un cours, un travail dirigé (TD), un travail pratique (TP) et un test. Nous avons associé à chaque situation la ressource nécessaire sous forme de texte, diapositives, vidéo ou audio comme illustré dans la Table 6.1. Nous avons fixé pour chaque ressource la note maximale  $M_{R_j} = 5$  qui est une note maximale complète caractérisant chaque ressource  $R_j$  dans une situation  $S_i$ , avec  $j = \{1, \dots, m_i\}$  et  $i = \{1, \dots, 3\}$ . Les apprenants ont suivi l'enchaînement des situations prévu du scénario pédagogique et ont été évalués sur leur degré d'assimilation des connaissances dans chacune des situations. Le résultat de l'évaluation des apprenants est détaillé dans la table 6.5 et présenté dans le graphique de la Figure 6.2.

TABLE 6.1 – Scénario de situations avec leurs ressources

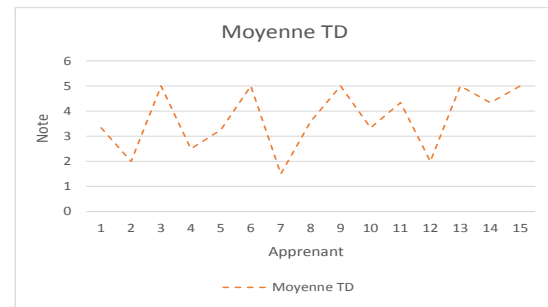
Situation	Objectif global	Objectif local	Ressource
Cours	Apprendre les principes théoriques de manipulation de tables	Création	R1= text
		Suppression	R2 = Slides
		Modification	R3= Video
Test cours	Épreuve permettant d'évaluer l'acquisition des connaissances de l'apprenant en situation de cours	Création	R1= text
		Suppression	R2 = Text
		Modification	R3= Text
Travaux dirigés	Exercer l'apprenant sur les concepts syntaxiques de la création, la suppression et modification de tables	Création	R1= text
		Suppression	R2 = text
		Modification	R3= text
Travaux pratiques	Écrire correctement les concepts syntaxiques des instructions du langage de définition de données (LDD)	Création	R1= text
		Suppression	R2 = text
		Modification	R3= text
Test	Examen ou épreuve permettant d'évaluer l'acquisition des connaissances de l'apprenant sur la manipulation de tables	Création	R1= text
		Suppression	
		Modification	

Le graphique de la Figure 6.2 présente les résultats d'évaluation en termes de moyenne de cours, TD, TP, test et moyenne générale en fonction de chaque apprenant.

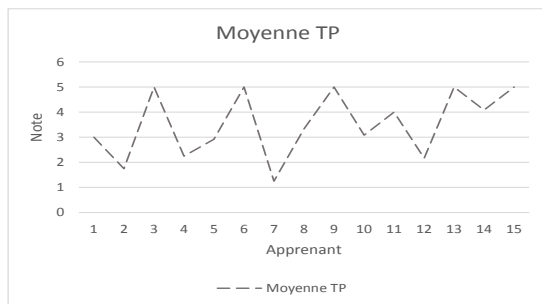
La table 6.2 résume les résultats des apprenants en situation de cours avec la formation traditionnelle. Les moyennes sont calculées en utilisant la formule classique, qui est la somme



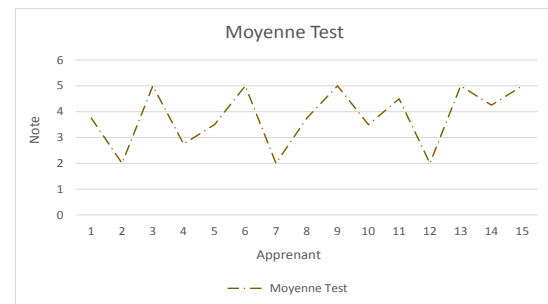
6.2.1 Moyenne de cours



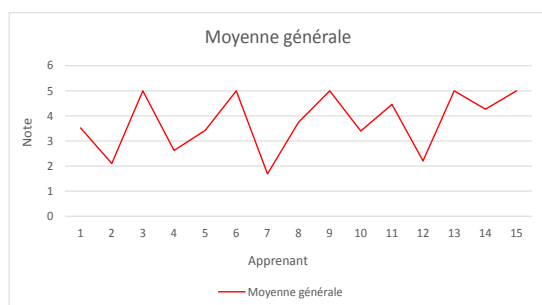
6.2.2 Moyenne de TD



6.2.3 Moyenne de TP



6.2.4 Moyenne de test



6.2.5 Moyenne générale

FIGURE 6.2 – Résultats des apprenants en formation traditionnelle.

des notes divisée par le nombre de notes. Dans la figure 6.2, la courbe de moyenne de cours des apprenants est présentée en trait continu bleu.

La figure 6.2 illustre les résultats de l'évaluation en termes de moyenne de cours, TD, TP, moyenne de test et moyenne globale de chaque apprenant. Nous remarquons que la plus petite moyenne obtenue est de 1,68 et que la meilleure est de 5. Les statistiques pour chaque situation en termes de minimum, maximum, moyenne de groupe et écart-type sont détaillées dans le tableau 6.4.

D'après l'analyse des résultats obtenus de la table 6.5 des situations de chaque apprenant, nous constatons que les apprenants qui ont réussi la situation de cours ont aussi réussi sans aucun problème les autres situations et dans le cas contraire l'inverse s'est produit. Pour cela, nous nous intéressons à une pré-détection de la faiblesse d'un apprenant dans une situation quelconque afin de le guider et de lui proposer une situation adaptée à son niveau et à ses

TABLE 6.2 – Scores obtenus pour la situation de cours dans la formation traditionnelle.

Apprenant	Création	Suppression	Modification	Moyenne
1	5	4	3	4
2	3	2	3	2,66
3	5	5	5	5
4	4	3	2	3
5	4	4	4	4
6	5	5	5	5
7	1	2	3	2
8	4	4	5	4,33
9	5	5	5	5
10	4	3	4	3,66
11	5	5	5	5
12	3	3	2	2,66
13	5	5	5	5
14	4,5	3,75	5	4,41
15	5	5	5	5
Moyenne	4,16	3,91	4,07	4,05

TABLE 6.3 – Statistiques des résultats obtenus dans la formation traditionnelle.

Situation	Min	Max	Moyenne	Écart-type
Cours	2	5	4,05	1,03
TD	1,5	5	3,68	1,25
TP	1,25	5	3,52	1,31
Test	2	5	3,8	1,16

performances réelles. Pour ce faire, nous nous basons sur les traces produites durant sa session d'apprentissage. Les traces considérées sont la durée passée par l'apprenant dans une situation d'apprentissage, le nombre de tentatives effectuées pour répondre à une question de l'examen et la combinaison des deux traces pour adapter les situations d'apprentissage. L'application de notre approche de scénarisation adaptée fondée sur les traces d'apprentissages est présentée dans ce qui suit.

TABLE 6.4 – Statistiques des valeurs obtenues dans l'apprentissage classique

Situation	Min	Max	Moyenne	Écart-type
Cours	2	5	4,05	1,03
TD	1,5	5	3,68	1,25
TP	1,25	5	3,52	1,31
Test	2	5	3,8	1,16

TABLE 6.5 – Les résultats obtenus en apprentissage classique

Apprenant	Situation de cours				Situation de TD				Situation de TP				Test	Moyenne générale
	Création	Suppression	Modification	Moyenne cours	Création	Suppression	Modification	Moyenne TD	Création	Suppression	Modification	Moyenne TP		
1	5	4	3	4	5	3	2	3,3	5	3	1	3	3,8	3,5
2	3	2	3	2,6	2	2	2	2	2	1,8	1,5	1,8	2	2,1
3	5	5	5	5	5	5	5	5	5	5	5	5	5	5
4	4	3	2	3	3	2	2,5	2,5	2,5	2	2,3	2,25	2,8	2,6
5	4	4	4	4	4	3	2,8	3,3	3,8	2,5	2,5	2,9	3,5	3,4
6	5	5	5	5	5	5	5	5	5	5	5	5	5	5
7	1	2	3	2	1	1,5	2	1,5	1	1	1,8	1,25	2	1,7
8	4	4	5	4,3	3	3,8	4	3,6	2,8	3,5	3,8	3,3	3,8	3,8
9	5	5	5	5	5	5	5	5	5	5	5	5	5	5
10	4	3	4	3,7	4	3	3	3,3	3,8	3	2,5	3,1	3,5	3,4
11	5	5	5	5	5	4	4	4,3	4,5	4	3,5	4	4,5	4,5
12	3	3	2	2,7	3	2	1	2	3	2,5	1	2,2	2	2,2
13	5	5	5	5	5	5	5	5	5	5	5	5	5	5
14	4,5	3,75	5	4,4	4	4	5	4,3	4	3,8	4,5	4,1	4,25	4,27
15	5	5	5	5	5	5	5	5	5	5	5	5	5	5

## 6.4 Expérimentation (2) : Apprentissage avec une scénarisation adaptée

Dans nos expérimentations des approches d'évaluation des apprenants et d'adaptation de ressources et/ou de situation d'apprentissage, nous avons exploité les résultats obtenus pour chaque apprenant dans chaque situation pour calculer les parcours d'apprentissage. Pour ce faire, les étapes suivantes ont été effectuées :

- Réglage des paramètres : dans l'évaluation des apprenants, nous avons pris en compte les traces laissées après chaque situation d'apprentissage. Nous définissons des paramètres différents pour chaque ressource  $R_j$  dans la situation  $S_i$ , avec  $j = \{1, \dots, m_i\}$  et  $i = \{1, \dots, n\}$  comme suit :
- $D_{R_j} = 50$  min pour le cours,  $D_{R_j} = 30$  min pour le TD,  $D_{R_j} = 30$  min pour le TP,  $D_{R_j} = 20$  min pour le Test, qui est la durée optimale pour compléter  $R_j$  ;
- le nombre de tentatives autorisées est  $T_{R_j} = 3$  ;
- trois niveaux de difficulté des ressources dans une situation  $\alpha_1 = 0,2$ ,  $\alpha_2 = 0,5$  et  $\alpha_3 = 0,8$ .
- Règles d'adaptation : les résultats obtenus à partir de nos modèles d'évaluation basés sur les traces, exprimés par les modèles (3.3), (3.1) et (3.2) ont été utilisés respectivement pour adapter les ressources et/ou les situations d'apprentissage en utilisant des règles d'inférences. Par exemple, la règle 6.4.1 permet une adaptation de la situation à l'apprenant en fonction de la trace de tentative, la règle 6.4.2 permet une adaptation de la situation en fonction de la trace temporelle et la règle 6.4.4 permet une adaptation de la situation basée sur la combinaison des traces susmentionnées ;
- Base de données ontologique : est une base de connaissances des situations et des ressources décrivant les activités d'apprentissage comme les contenus d'enseignement stockés dans une base de données, et une modélisation ontologique des concepts nécessaires en termes de classes et propriétés de notre domaine. L'ontologie permet d'exprimer des règles logiques sur lesquelles un raisonnement est construit pour adapter les ressources et les situations ;
- Résultats : basé sur les trois types de traces, les résultats de situations d'apprentissage sont donnés comme suit :
  - la figure 6.3 illustre les résultats des apprenants en situation de cours en fonction des traces de tentatives dans les trois niveaux de difficultés ;



- la figure 6.5 concerne les résultats du cours basés sur la trace temporelle ;
- la figure 6.7 montre les résultats du cours basés sur la combinaison des deux traces. Dans ces graphiques, la courbe de moyenne des apprenants est présentée en pointillés rouges.

Dans le paragraphe 6.4.1, nous détaillons l'expérimentation en apprentissage avec une scénarisation adaptée à base de la trace de tentative. Dans le paragraphe 6.4.2 la seconde expérimentation en apprentissage adapté à base de la trace temporelle. L'expérimentation à base de la combinaison des deux traces est détaillée dans le paragraphe 6.4.3.

### 6.4.1 Apprentissage avec une scénarisation adaptée à base de la trace de tentative

Après la collecte des traces obtenues de la session d'apprentissage, nous avons utilisé le modèle d'évaluation basé sur la trace de tentative, exprimés par le (modèle 3.2) pour calculer les moyennes de notes des apprenants obtenues durant la session de cours. Nous avons fait la somme des notes obtenues pour les différentes valeurs de  $\alpha$  divisé par le nombre de notes de différents niveaux de difficultés des ressources, comme détaillé dans la table 6.6. Les résultats de calcul ont été injectés dans la base de données ontologique pour adapter les situations d'apprentissage et les ressources associées aux apprenants.

La règle 6.4.1 présente l'adaptation de la situation d'apprentissage basée sur la trace de tentative, en cas d'échec de l'apprenant dans la situation actuelle.

```
1. EXECUTE SEM_APIS.CREATE_RULEBASE('attemptsAdaptedSituation_rb');
2. INSERT INTO mdsys.sem_r_adaptedSituation_rb
3. VALUES('courseAdaptedSituation_rule',
'(?1Obj1 rdf:type os:Learner)(?sObj3 rdf:type os:Lecture)(?1Obj1 os:doSituation ?sObj3)(?1Obj1 os:hasMark ?mark
) (?1Obj1 os:hasAttempts ?attempts)',
4. '(mark < 5)and(attempts > 3)',
5. '(?1Obj1 os:courseAdaptedSituation ?sObj3)',
6. SEM_ALIASES(SEM_ALIAS('os','http://13i.univ-lr.fr/eadaptOnto')));
```

Code 6.4.1 – Création de la règle à base de tentative

TABLE 6.6 – Résultats de Cours à base de la trace de tentative

		<b>Création</b>	<b>Suppression</b>	<b>Modification</b>	
<b>Apprenant</b>	<b>T réalisé</b>	$\alpha = 0,2$	$\alpha = 0,5$	$\alpha = 0,8$	<b>Moyenne</b>
1	1	5	5	5,00	5,00
2	4	4,68	4,23	3,83	4,25
3	2	5,00	5,00	5,00	5,00
4	6	4,09	3,03	2,25	3,12
5	3	5,00	5,00	5,00	5,00
6	5	4,38	3,58	2,93	3,63
7	2	5,00	5,00	5,00	5,00
8	4	4,68	4,23	3,83	4,25
9	1	5,00	5,00	5,00	5,00
10	2	5,00	5,00	5,00	5,00
11	3	5,00	5,00	5,00	5,00
12	2	5,00	5,00	5,00	5,00
13	4	4,68	4,23	3,83	4,25
14	1	5,00	5,00	5,00	5,00
15	3	5,00	5,00	5,00	5,00

Les résultats des apprenants obtenus après adaptation des situations sont illustrés dans la Figure 6.3 qui présente les résultats d'évaluation en termes de la moyenne de cours pour les trois ressources utilisées en apprentissage du LDD SQL avec les trois niveaux de difficulté suivants : 0,2 pour un niveau débutant, 0,5 pour intermédiaire et 0,8 pour difficile. L'évaluation a été effectuée en fonction de chaque apprenant et de son nombre de tentatives réalisés. Selon une première analyse des résultats obtenus, nous remarquons une amélioration des résultats lorsque l'on met les apprenants sous une pression de limitation du « nombre de tentatives » en comparaison avec ceux obtenus dans l'approche classique comme l'illustre la Table 6.7 et la figure 6.4.

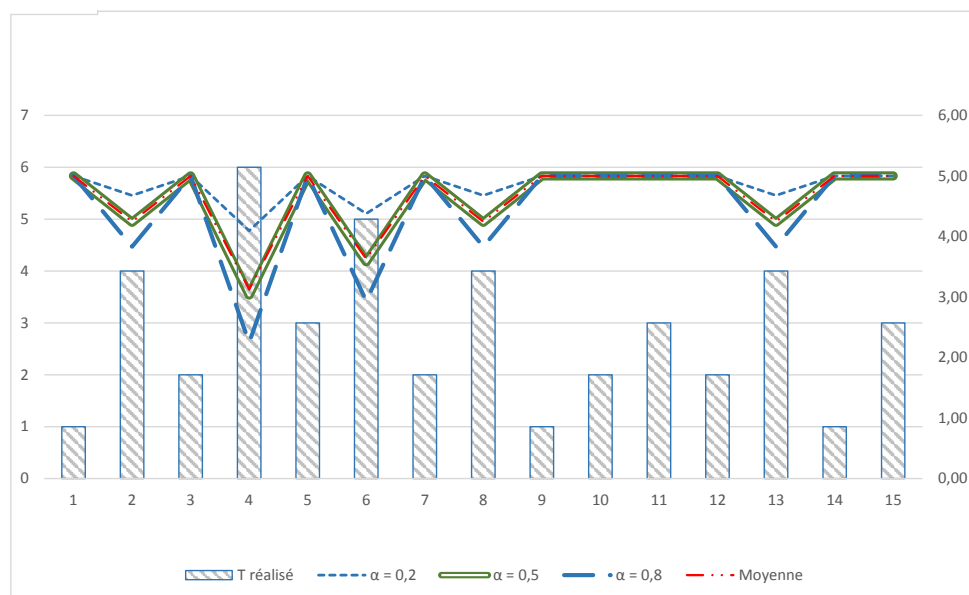


FIGURE 6.3 – Résultats de cours à base de la trace de tentative.

TABLE 6.7 – Statistiques des résultats de cours obtenus sans et avec adaptation à base de la trace de tentatives

Situation	Min	Max	Moyenne	Écart-type
Cours sans adaptation	2	5	4,05	1,03
Cours avec adaptation	3,12	5	4,63	0,60
Variation	56%	—	14,32%	41,75%

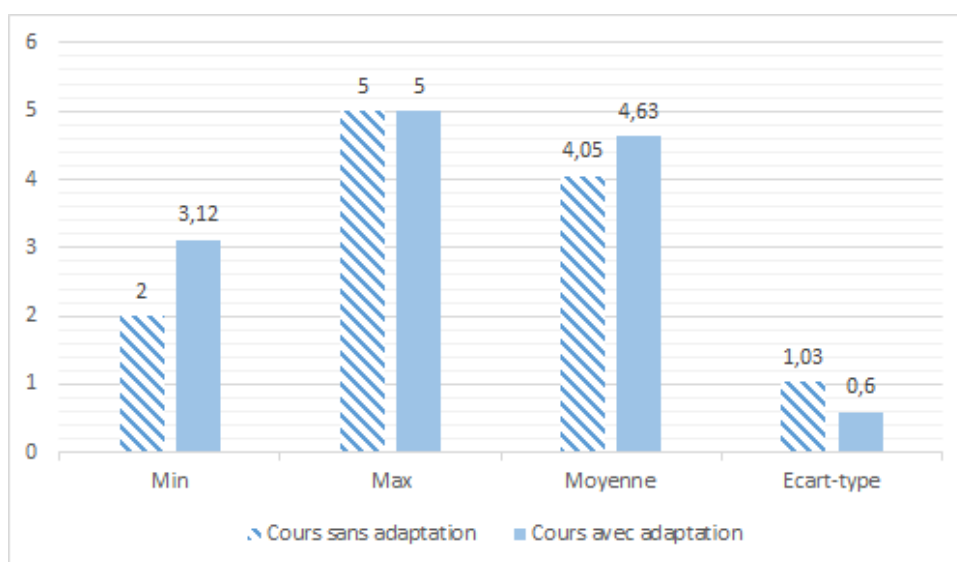


FIGURE 6.4 – Comparaison des résultats cours des deux groupes avec et sans adaptation.

Selon la Table 6.7, nous remarquons que la note minimale obtenue a augmenté de 56%, la moyenne du groupe a augmenté de plus de 14% et l'écart-type a baissé de 41.75% dans le

cas de l'adaptation de l'apprentissage par rapport à l'apprentissage classique. Ce qui indique une amélioration de l'acquisition pédagogique du groupe en plus de l'homogénéisation de son niveau (diminution de l'écart-type avec augmentation de la moyenne).

### 6.4.2 Apprentissage avec une scénarisation adaptée à base de la trace temporelle

Le graphique de la Figure 6.5 présente les résultats d'évaluation en termes de la moyenne de cours pour les trois ressources utilisées de la table 6.2. L'évaluation a été effectuée en utilisant le modèle 3.1 pour chaque apprenant et en fonction de la durée passée dans la situation d'apprentissage, comme détaillé dans la table 6.8. Les résultats de calcul ont été injectés dans la base de données ontologique pour adapter les situations d'apprentissage et les ressources associées aux apprenants qui ont réussi ou échoué la situation d'apprentissage.

La règle ?? présente l'adaptation de la situation d'apprentissage basée sur la trace temporelle, en cas d'échec de l'apprenant dans la situation actuelle.

```
1. EXECUTE SEM_APIS.CREATE_RULEBASE('durationAdaptedSituation_rb');
2. INSERT INTO mdsys.semr_adaptedSituation_rb
3. VALUES('courseAdaptedSituation_rule',
'(?1Obj1 rdf:type os:Learner)(?sObj3 rdf:type os:Lecture)(?1Obj1 os:doSituation ?sObj3)(?1Obj1 os:hasMark ?mark
)(?1Obj1 os:hasDuration ?duration)',
4. '(mark < 5)and(duration >50)',
5. '(?1Obj1 os:courseAdaptedSituation ?sObj3)',
6. SEM_ALIASES(SEM_ALIAS('os','http://l3i.univ-lr.fr/eadaptOnto')));
```

Code 6.4.2 – Création de la règle à base de la durée d'interaction

TABLE 6.8 – Résultats de test de Cours à base de la trace temporelle

		Création	Suppression	Modification	
Apprenant	Durée	$\alpha = 0,2$	$\alpha = 0,5$	$\alpha = 0,8$	Moyenne
1	20	5	5	5,00	5,00
2	50	4,38	3,58	2,93	3,63
3	40	4,68	4,23	3,83	4,25
4	62	4,04	2,93	2,13	3,03
5	50	4,38	3,58	2,93	3,63
6	40	4,68	4,23	3,83	4,25
7	45	4,52	3,89	3,35	3,92
8	60	4,09	3,03	2,25	3,12
9	30	5,00	5,00	5,00	5,00
10	40	4,68	4,23	3,83	4,25
11	40	4,68	4,23	3,83	4,25
12	50	4,38	3,58	2,93	3,63
13	60	4,09	3,03	2,25	3,12
14	30	5,00	5,00	5,00	5,00
15	55	4,23	3,30	2,57	3,73

La courbe des moyennes de la figure 6.5 est variable selon la moyenne obtenue de chaque apprenant et pour chaque niveau de difficulté de la ressource utilisée. Selon une première analyse des résultats obtenus, nous remarquons une amélioration des résultats lorsque l'on met les apprenants sous une pression « temporelle » en comparaison avec ceux obtenus dans l'approche classique.

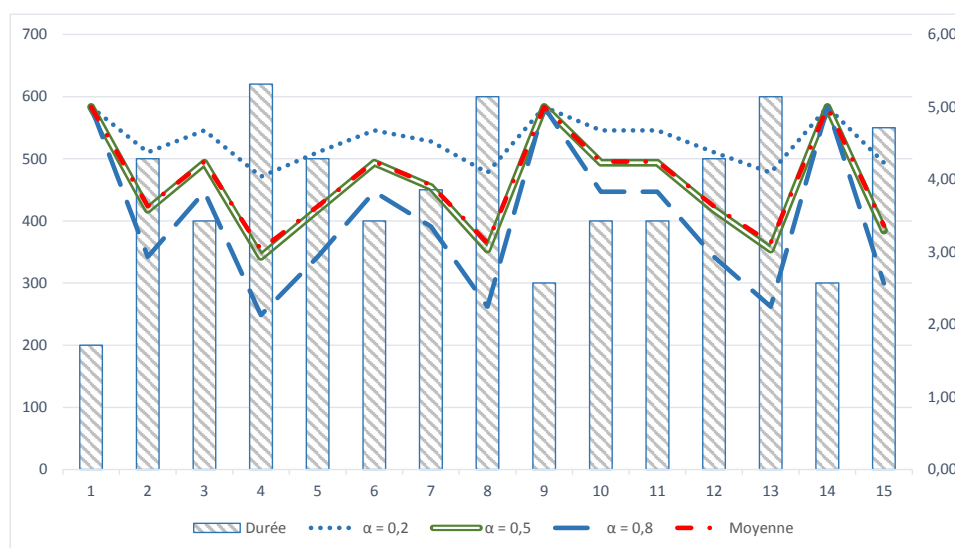


FIGURE 6.5 – Résultats de cours à base de la trace temporelle.

Selon les résultats de la Table 6.9, nous remarquons que la note minimale obtenue a augmenté de 50%, la moyenne du groupe a augmenté de 1,24% et l'écart-type a baissé de 33,98% dans le cas de l'adaptation de l'apprentissage par rapport à l'apprentissage classique. Ce qui indique une amélioration de l'acquisition pédagogique du groupe en plus de l'homogénéisation de son niveau (diminution de l'écart-type avec une légère augmentation de la moyenne).

TABLE 6.9 – Statistiques des résultats de cours obtenus sans et avec adaptation à base de la trace temporelle

Situation	Min	Max	Moyenne	Écart-type
Cours sans adaptation	2	5	4,05	1,03
Cours avec adaptation	3,03	5	4,10	0,68
Variation	50%	—	1,24%	33,98%

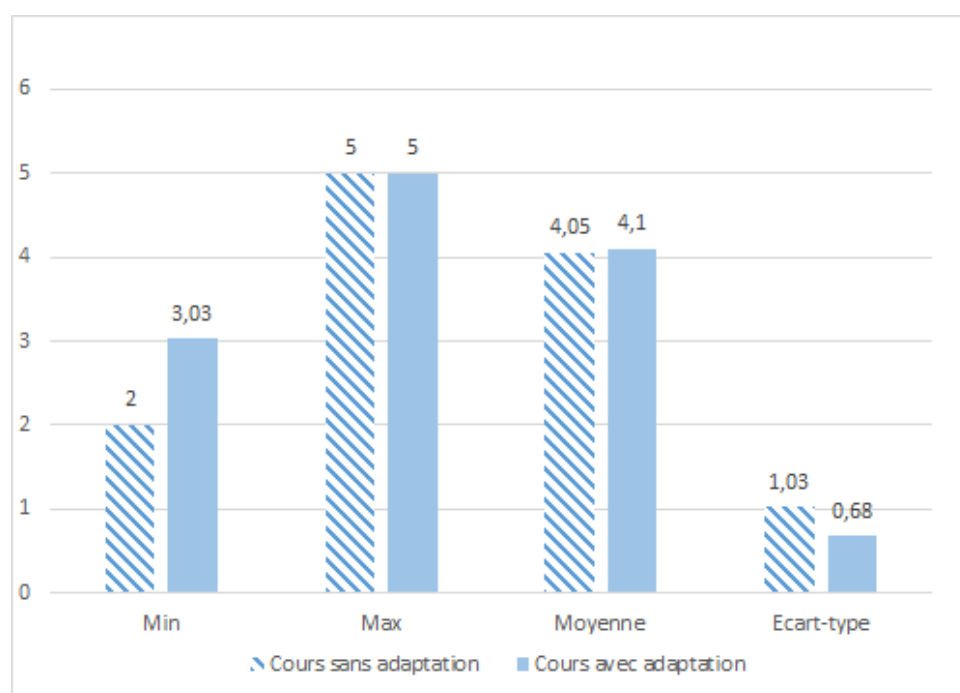


FIGURE 6.6 – Comparaison des résultats cours des deux groupes avec et sans adaptation.

### 6.4.3 Apprentissage avec une scénarisation adaptée à base de la combinaison de la trace temporelle et de tentative

Le graphique de la Figure 6.7 présente les résultats d'évaluation en termes de la moyenne de cours pour les trois ressources utilisées. L'évaluation a été effectuée en fonction de chaque apprenant, et la combinaison du nombre de tentatives et la durée passée dans la situation d'apprentissage, comme détaillé dans la table 6.10 et illustré sur la figure 6.7. Nous remarquons une amélioration des résultats lorsque l'on met les apprenants sous une pression combinée

« du nombre de tentatives et temporelle » en comparaison avec ceux obtenus dans l'approche classique.

TABLE 6.10 – Résultats de Cours à base de la combinaison de la trace temporelle et de tentative

			Création	Suppression	Modification	
Apprenant	Durée	T réalisé	$\alpha = 0,2$	$\alpha = 0,5$	$\alpha = 0,8$	Moyenne
1	20	1	5,00	5,00	5,00	5,00
2	50	4	4,09	3,03	2,25	3,12
3	40	2	4,29	3,41	2,71	3,47
4	62	6	3,31	1,78	0,96	2,01
5	50	3	5,00	5,00	5,00	5,00
6	40	5	4,09	3,03	2,25	3,12
7	45	2	5,00	5,00	5,00	5,00
8	60	4	3,83	2,57	1,72	2,71
9	30	1	5,00	5,00	5,00	5,00
10	40	2	5,00	5,00	5,00	5,00
11	40	3	5,00	5,00	5,00	5,00
12	50	2	5,00	5,00	5,00	5,00
13	60	4	3,83	2,57	1,72	2,71
14	30	1	5,00	5,00	5,00	5,00
15	55	3	5,00	5,00	5,00	5,00

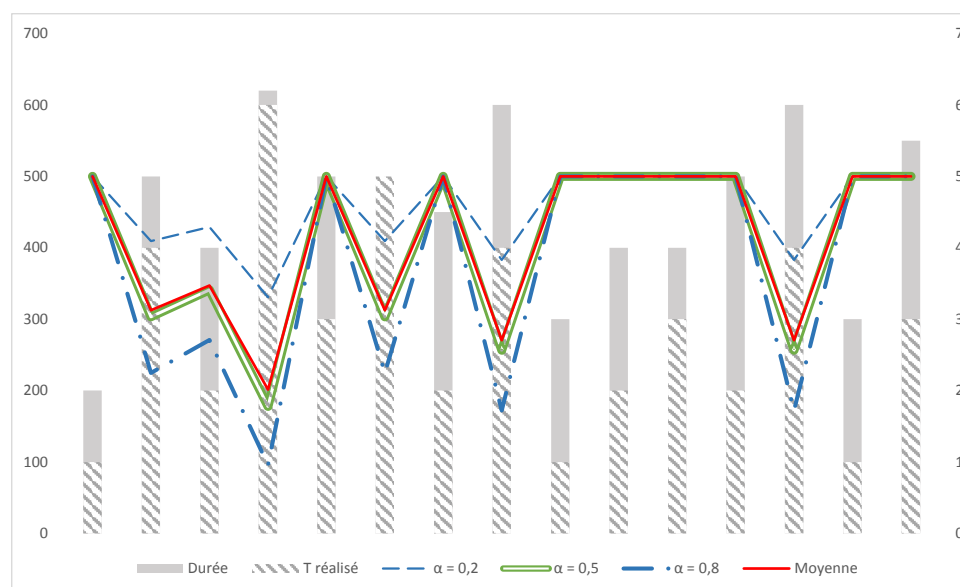


FIGURE 6.7 – Résultats de cours à base de la combinaison de la trace temporelle et de tentative.

Après avoir collecté les données et les résultats obtenus à partir du modèle d'évaluation basé sur la combinaison de traces temporelles et de traces exprimé par le (modèle 3.3), nous

l'avons injecté dans notre base de données ontologique, pour activer la règle de raisonnement du code 6.4.3 qui renvoie la ressource adaptée aux apprenants qui ont échoué dans la situation d'apprentissage. Quant aux apprenants qui ont réussi la situation d'apprentissage, la règle 6.4.5 renvoie leurs prochaine situation adaptée à exécuter.

```

1. EXECUTE SEM_APIS.CREATE_RULEBASE('AttemptsDurationAdaptedResource_rb');
2. INSERT INTO mdsys.semr_AttemptsDurationAdaptedResource_rb
3. VALUES('courseAdaptedResource_rule',
'(?lObj1 rdf:type os:Learner)(?sObj2 rdf:type os:Situation)(?rObj3 rdf:type os:Ressource)(?lObj1 os:
doSituation ?sObj2)(?lObj1 os:exploitResource ?rObj3)(?lObj1 os:hasMark ?mark)(?lObj1 os:hasDuration ?
duration)(?lObj1 os:hasAttempts ?attempts)',
4. '(mark < 5)and(duration >50)or(attempts > 3)',
5. '(?lObj1 os:courseAdaptedResource_rule ?rObj3)',
6. SEM_ALIASES(SEM_ALIAS('os','http://13i.univ-lr.fr/eadaptOnto')));

```

Code 6.4.3 – Création de la règle d'adaptation de la ressource à base de la durée et le nombre de tentatives

Nous appliquons par la suite les règles d'adaptation des situations d'apprentissage comme suit :

### Règle d'adaptation de la situation d'apprentissage à base de la combinaison des deux traces (cas d'échec de l'apprenant dans la situation actuelle)

```

1. EXECUTE SEM_APIS.CREATE_RULEBASE('AttemptsDurationAdaptedSituation_rb');
2. INSERT INTO mdsys.AttemptsDurationAdaptedSituation_rb
3. VALUES('courseAdaptedSituation_rule',
'(?lObj1 rdf:type os:Learner)(?sObj3 rdf:type os:Lecture)(?lObj1 os:doSituation ?sObj3)(?lObj1 os:hasMark ?mark
)(?lObj1 os:hasDuration ?duration)(?lObj1 os:hasAttempts ?attempts)',
4. '(mark < 5)and(duration >50)and(attempts > 3)',
5. '(?lObj1 os:courseAdaptedSituation ?sObj3)',
6. SEM_ALIASES(SEM_ALIAS('os','http://13i.univ-lr.fr/eadaptOnto')));

```

Code 6.4.4 – Création de la règle d'adaptation de la situation à base de la durée et le nombre de tentatives

Les apprenants concernés par l'adaptation ont refait les situations adaptées. Les résultats obtenus sont illustrés par la Figure 6.8.



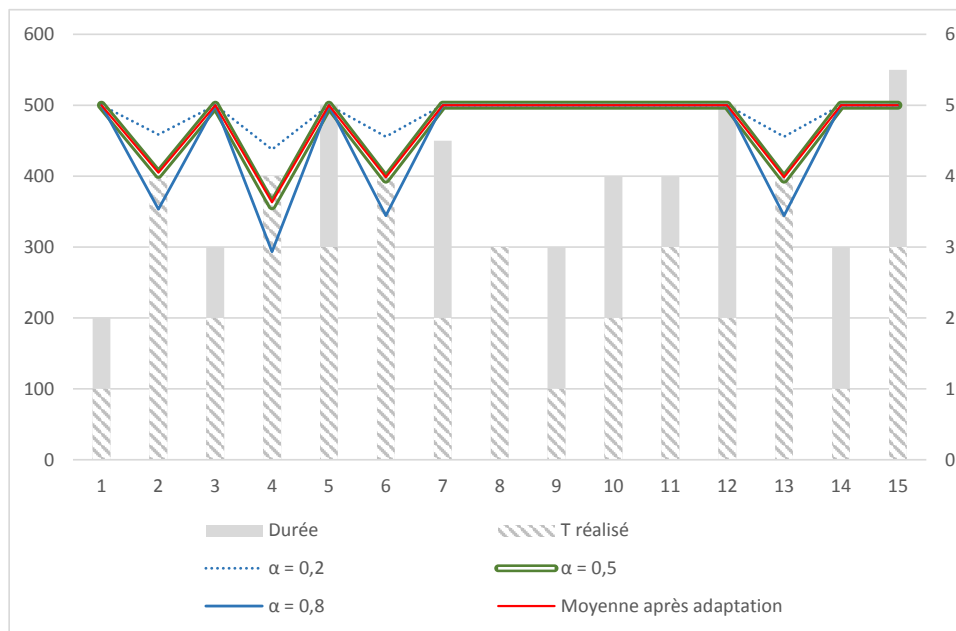


FIGURE 6.8 – Résultats du cours après adaptation à base de la combinaison de la trace temporelle et de tentative

### Règle d'adaptation de la situation d'apprentissage à base de la trace de note (cas du succès de l'apprenant dans la situation actuelle)

```

1. EXECUTE SEM_APIS.CREATE_RULEBASE('noteAdaptedSituation_rb');
2. INSERT INTO mdsys.semr_noteAdaptedSituation_rb
3. VALUES('DWadaptedSituation_rule',
4. '(?lObj1 rdf:type os:Learner)(?sObj3 rdf:type os:Lecture)(?sObj4 rdf:type os:DirectedWork)(?rObj4 rdf:type
   os:Ressource)(?sObj4 os:useResource ?rObj4)(?lObj1 os:doSituation ?sObj3)(?lObj1 os:hasMark ?mark)(?lObj1
   os:hasDuration ?duration)(?lObj1 os:hasAttempts ?attempts)',
5. '(mark = 5)and(duration <50)or(attempts < 3)',
6. '(?lObj1 os:DWadaptedSituation ?sObj4)',
7. SEM_ALIASES(SEM_ALIAS('os','http://13i.univ-lr.fr/eadaptOnto')));

```

Code 6.4.5 – Création de la partie impérative de la règle learnerSuccesGoNextSituation

Selon les statistiques de la Table 6.11 et la figure 6.9, nous remarquons que la note minimale obtenue a augmenté de 81,5%, la moyenne du groupe a augmenté de plus de 16,30% et l'écart-type a baissé de 51,46% dans le cas de l'adaptation de l'apprentissage par rapport à l'apprentissage classique. Ce qui indique une amélioration de l'acquisition pédagogique du groupe en plus de l'homogénéisation de son niveau.

TABLE 6.11 – Statistiques des valeurs de cours obtenues sans et avec adaptation à base de la trace de temporelle et de tentatives

Situation	Min	Max	Moyenne	Écart-type
Cours sans adaptation	2	5	4,05	1,03
Cours avec adaptation	3,63	5	4,71	0,50
Variation	81,5%	—	16,30%	51,456%

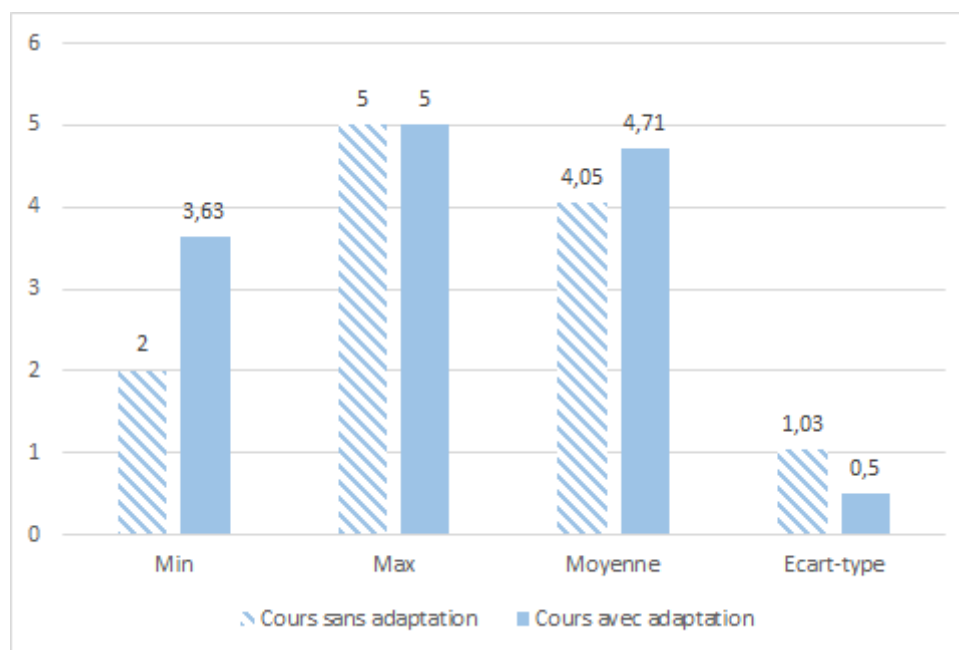


FIGURE 6.9 – Comparaison des résultats cours des deux groupes avec et sans adaptation.

#### 6.4.4 Résultats obtenus par les situations TD, TP et Test

Dans ce qui suit, nous avons effectué les mêmes expérimentations sur les travaux dirigés (TD), les travaux pratiques (TP) et la situation de test. Nous avons calculé la moyenne globale en tenant compte de la combinaison des deux traces de tentatives et temporelles. Les apprenants qui n'ont pas atteint leur objectif dans une situation de formation donnée sont obligés de refaire une situation adaptée à leur niveau d'acquisition au cours du processus d'apprentissage.

Nous avons suivi le même processus d'adaptation pour le scénario d'apprentissage proposé dans le tableau 6.1. Les statistiques des résultats obtenus des différentes situations sont détaillés dans le tableau 6.12, et illustrés dans les graphiques de la figure 6.10.

Nous remarquons que le groupe d'apprenants qui ont suivi un apprentissage adapté basé sur les traces a amélioré leur niveau et obtenu de meilleurs résultats par rapport à ceux de l'apprentissage traditionnel. Cette amélioration a un impact positif sur la moyenne globale des apprenants soumis à l'adaptation, comme le montre la figure 6.10.

TABLE 6.12 – Statistiques des valeurs de TD, TP, Test et moyenne obtenues sans et avec adaptation

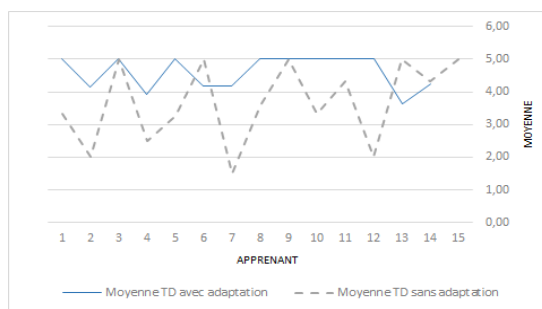
Situation	Min	Max	Moyenne	Écart-type
TD sans adaptation	1,5	5	3,68	1,25
TD avec adaptation	3,63	5	4,62	0,50
Variation	+142%	-	+25,54%	-60 %
TP sans adaptation	1,25	5	3,52	1,31
TP avec adaptation	3,63	5	4,63	0,49
Variation	+190,4%	-	+31,53%	-62,60%
Test sans adaptation	2	5	3,8	1,16
Test avec adaptation	4,28	5	4,74	0,38
Variation	+114%	-	+24,74%	-67,24%
Moyenne générale sans adaptation	1,69	5	3,76	1,18
Moyenne générale avec adaptation	3,86	5	4,67	0,38
Variation	+128,40%	-	+24,20	-67,80%

D'après le tableau 6.12, le score minimum obtenu en TD, TP et Test a augmenté de 142%, 190% et 114% respectivement, la moyenne du groupe a aussi augmentée de 25,54%, 31,53%, et 24,74% respectivement, et l'écart type a diminué de 60%, 62,60% et 67,24% respectivement dans le cas de la formation adaptée. De plus, dans la figure 6.10 la moyenne minimale obtenue a augmenté de 128,40%, la moyenne du groupe a augmenté de plus de 24,20% et l'écart-type a diminué de 67,80 dans le cas la formation adaptée. Cela indique une amélioration de l'acquisition pédagogique du groupe en plus de l'homogénéisation de son niveau.

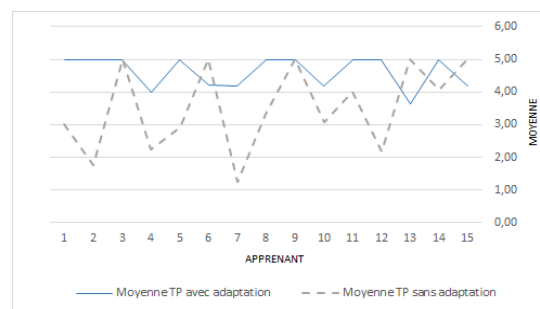
## 6.5 Bilan

À travers les résultats obtenus des deux phases d'expérimentation, nous avons noté une amélioration des performances pédagogiques des apprenants qui ont suivi une formation adaptée en comparaison à ceux qui ont suivi une formation classique.

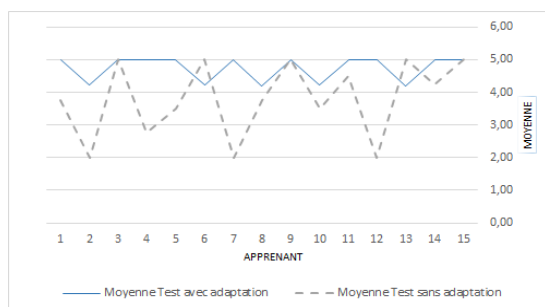
Toutefois, en analysant les résultats de plus près, nous remarquons que cinq apprenants n'ont pas amélioré leurs résultats même après adaptation. Ce constat nous renseigne sur l'existence d'autres facteurs pédagogiques que notre approche n'a pas su capter dans le processus d'apprentissage. Pour comprendre ces situations d'échec, nous avons demandé à ces apprenants de répondre à un questionnaire au sujet de leur expérience d'apprentissage. Le problème principal est le stress imposé par la limitation temporelle et le nombre de tentatives. Ces facteurs ont diminué la motivation de quelques apprenants qui ne sont pas habitués à ce genre d'apprentissage. Il est alors important de savoir comment paramétrer les modèles afin de réduire cet effet indésirable de stress.



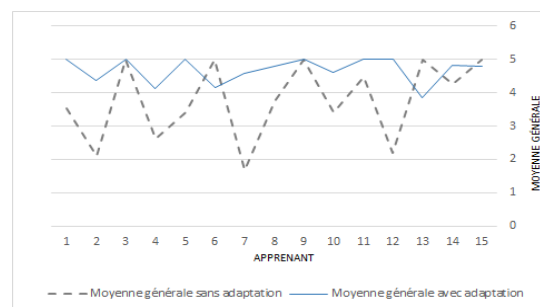
6.10.1 Résultats de TD



6.10.2 Résultats de TP



6.10.3 Résultats de Test



6.10.4 Résultats de moyennes générales

FIGURE 6.10 – Comparaison des résultats TD, TP, Test et moyenne générale du groupe avec et sans adaptation.

Par ailleurs, il est aussi nécessaire de généraliser les résultats obtenus à travers de tests effectués sur un plus grand nombre d'étudiants, mais aussi en variant les différents types d'apprentissage.

## 6.6 Conclusion

Ce chapitre présente l'évaluation et la validation de nos modèles d'évaluation et le processus d'adaptation à base de traces qui ont été détaillés dans les chapitres 3 et 4.

Pour ce faire, nous avons effectué une expérimentation sur deux groupes composés de quinze apprenants chacun. Le premier groupe a suivi une formation classique et le deuxième groupe a suivi une formation adaptée basée sur les traces d'apprentissage.

Pour chaque situation d'apprentissage du deuxième groupe, nous avons injecté dans notre base de données ontologique à base de règles les résultats d'apprentissage des apprenants obtenus et calculés par nos modèles d'évaluation à base de traces. Cette phase nous a permis d'obtenir la situation ou la ressource adaptée au niveau de l'apprenant.

À la fin de la session d'apprentissage, nous avons comparé les résultats obtenus des deux groupes et nous avons constaté que les apprenants qui ont suivi une formation adaptée ont

---

amélioré leur niveau au fur et à mesure de l'adaptation d'une situation ou d'une ressource à leur performance par rapport à ceux qui ont suivi une classe classique.

# **Partie V**

## **Conclusion**



# Chapitre 7

## Conclusion Générale

Dans ce chapitre nous présentons une conclusion portant sur nos travaux de thèse et des perspectives de développements théoriques et pratiques futurs. Nous débutons par un bilan sur l'apport bibliographique, méthodologiques et expérimentaux. Nous présentons les avantages et les inconvénients de nos approches proposées. Enfin, nous finissons par des perspectives qui peuvent être un début à d'autres travaux de recherche dans le domaine d'évaluation de l'apprenant en ce qui concerne son acquisition des compétences et connaissances pédagogiques ainsi que dans le domaine de l'adaptation de la stratégie d'apprentissage suivant les besoins et objectifs pédagogiques.

### 7.1 Bilan des contributions et apports de la thèse

Les systèmes tutoriels intelligents (STI) est une famille particulière des systèmes interactifs adaptatifs. L'exécution adaptative dans les STI repose sur ses composants intelligents et en particulier sur ses modèles (modèle de l'apprenant, modèle pédagogique et modèle d'expert). L'exécution prend en considération le profil et les caractéristiques de l'apprenant pour ajuster les formations suivant ses objectifs et ses besoins. Dans ce contexte, l'objectif principal de cette thèse est d'évaluer l'activité de l'apprenant suivant ses productions et ses caractéristiques dynamiques durant l'interaction avec une situation d'apprentissage. Le résultat de cette évaluation nous a permis par la suite d'adapter la ressource et/ou la stratégie d'apprentissage suivant ses compétences et ses performances. Donc, l'objectif est d'intégrer un processus d'évaluation de l'apprenant et d'adaptation de ressource et de stratégie pédagogique au système STI en se basant sur les traces.

Les travaux réalisés durant cette thèse ont donné lieu aux apports bibliographiques, méthodologiques et expérimentaux que nous détaillons ci-dessous.



### 7.1.1 Apports bibliographiques

Nos apports de recherches ont permis d'établir un état de l'art sur les systèmes tutoriels intelligents (STI) qui représente notre champ d'étude pour la mise en œuvre de notre approche d'évaluation à base de traces et qui a servi par la suite à l'adaptation de la ressource et la stratégie d'apprentissage.

Dans le chapitre 1 nous avons défini les systèmes STI ainsi que leurs caractéristiques, leurs classifications et les composantes nécessaires : le modèle expert, le modèle de l'apprenant et le modèle pédagogique. Nous nous sommes basés sur ces derniers pour fonder nos travaux de recherches. Nous avons utilisé le modèle de l'apprenant pour construire notre modèle d'évaluation de l'apprenant en se basant sur les traces collectées et modélisées. Le modèle pédagogique nous a servi à appliquer notre modèle d'adaptation de ressource et/ou stratégie d'apprentissage qui est bâti sur nos modèles d'évaluation et de trace.

Le chapitre 2 a été consacré à un état de l'art sur les méthodes d'évaluation de l'apprenant existantes dans la littérature. Nous nous sommes concentrés sur les méthodes d'évaluation automatisées des apprenants. Par ailleurs, nous avons étudié l'évaluation dans les systèmes tutoriels intelligents (STI). Ainsi, le fait que nos objectifs portent sur l'adaptation, nous nous sommes orientés vers l'évaluation adaptée de l'apprenant dans les systèmes d'apprentissage en ligne.

Dans le même chapitre, nous avons effectué un état de l'art sur les systèmes d'adaptation dans les plateformes d'apprentissage en ligne en se concentrant sur les systèmes STI. Nous avons terminé le chapitre par un bilan dans lequel nous avons identifié les insuffisances dans les modèles d'évaluation et les systèmes d'adaptation pédagogique développés ou proposés dans la littérature, sur lesquels nous avons construit nos contributions.

Le chapitre 3 détaille nos contributions en ce qui concerne l'évaluation de l'apprenant fondée sur les traces obtenues durant une activité d'apprentissage. Tout d'abord, nous avons défini les concepts de traces, modèles de traces, modèles d'évaluation, scénario et situation.

Ces concepts nous ont permis la construction de trois modèles d'évaluation à base de traces, à savoir :

- modèle d'évaluation fondé sur la durée d'interaction ;
- modèle d'évaluation fondé sur le nombre de tentatives ;
- modèle d'évaluation combiné fondé sur la durée et le nombre de tentatives.

Par ailleurs, nous avons développé les composantes de notre processus d'évaluation à base de traces.

Le chapitre 4 présente nos contributions en ce qui concerne l'adaptation de la ressource et de la stratégie d'apprentissage à base de traces d'un système STI. Nous avons défini les

différentes composantes de notre architecture du système d'adaptation à base de traces. Nous avons proposé une description ontologique de notre domaine de formation à base de situations fondé sur les traces. Le processus de raisonnement sur la base ontologique est assuré par un algorithme qui nécessite l'interaction avec une base de données de traces.

Le chapitre 5 est dédié à l'implémentation de notre modèle d'évaluation et d'adaptation comme preuve de concept dans une ontologie à base de règles d'inférences. Enfin, le chapitre 6 présente l'évaluation et la validation des contributions au travers des expérimentations de nos approches d'évaluation de l'apprenant et de l'adaptation de la ressource et de la situation d'apprentissage sur une population d'étudiants en Licence Informatique (L2).

## **7.1.2 Apports méthodologiques**

### **7.1.2.1 Apports sur les traces**

**Quelles traces utilisons-nous ? Quel modèle de traces utilisons-nous ? Quelles données de traces sont-elles utiles à nos approches proposées ?**

À partir des sessions d'apprentissage des étudiants en apprentissage adapté, nous avons récupéré les traces produites par l'apprenant après son interaction avec les ressources dans une situation. Ces dernières sont observées, collectées, analysées et modélisées sous forme de m-trace par un système à base de trace (SBT). Le modèle de trace que nous avons défini est à base des deux traces, à savoir, la trace temporelle et la trace de tentative faites par un apprenant pour accomplir une situation d'apprentissage ou un objectif donné.

### **7.1.2.2 Apports sur l'évaluation**

Nous avons proposé un processus de notation de l'apprenant dans un système de STI et nous avons décrit les différents aspects pour la mise en œuvre de ce processus. Ce processus prend en considération l'exécution d'une ou de plusieurs situations conçues par le concepteur. À la fin de l'exécution de la situation, l'apprenant est évalué par un des modèles d'évaluation que nous avons proposés. En se basant sur sa performance en acquisition et ses résultats obtenus, le système peut lui proposer la prochaine situation adaptée à son niveau et à ses objectifs à atteindre.

**Quels sont les modèles à utiliser pour l'évaluation adaptée ?**

Après l'état de l'art effectué sur les modèles d'évaluation existants, nous avons développé un processus de notation qui prend en considération les caractéristiques dynamiques d'un profil apprenant. Il se base sur les étapes suivantes :

- le paramétrage de l'activité ou de la situation d'apprentissage ;

- observation des événements et traçage des interactions et actions de l'apprenant ;
- collecte, analyse et modélisation des traces ;
- évaluation de la réponse (correcte ou incorrecte) ;
- notation de l'apprenant ;
- adaptation de la ressource et/ou la situation d'apprentissage pour l'apprenant.

Nous avons proposé des modèles d'évaluation basés sur des fonctions de notation quantitatives qui exploitent les traces produites par un apprenant durant une situation d'apprentissage.

Le système suit trois logiques avant, pendant et après l'exécution d'une situation d'apprentissage.

1. Avant l'exécution de la situation : nous avons proposé un paramétrage de la situation d'apprentissage par le concepteur ou l'expert du domaine en fixant la note complète, la durée optimale et le nombre d'essais autorisés pour qu'un apprenant peu résoudre un problème donné. D'autres paramètres d'atténuation ont été proposés pour indiquer le niveau de difficulté de la ressource utilisée par l'apprenant. L'intérêt de ce paramétrage est l'attribution d'une ressource suivant le niveau de connaissances de l'apprenant ;
2. Pendant l'exécution d'une situation : la logique d'observation de l'activité est illustrée comme suit :
  - nous avons rattaché pour chaque événement un observateur qui nous permet de tracer toutes les interactions et actions pendant l'interaction de l'apprenant avec la plateforme ou le système ;
  - les traces produites sont collectées, analysées et enregistrées dans un système à base de traces (SBT). Ces dernières sont modélisées sous forme d'un modèle de trace (m-trace) adapté qui alimente nos modèles d'évaluation par la suite ;
  - la réponse de l'apprenant est vérifiée si elle est correcte ou fausse.
  - Après l'exécution d'une situation : processus d'évaluation et d'adaptation en se basant sur les traces produites est décrit par les étapes suivantes :
  - la notation du travail de l'apprenant est basée sur nos modèles d'évaluation à base de traces ;
  - à travers les résultats obtenus par l'apprenant nous pouvons déduire l'évolution en termes d'acquisition de connaissances en lui attribuant une ressource et/ou situation adaptée à son niveau de performance dans le processus d'apprentissage.

L'avantage de notre approche est l'adaptation de la ressource ou de la stratégie d'apprentissage suivant les performances d'acquisition de connaissances de l'apprenant.

### 7.1.3 Apports sur l'adaptation

Nous avons proposé une logique de pilotage de l'activité à base de traces. Le pilotage est décrit dans un processus d'adaptation de la ressource et/ou de la stratégie d'apprentissage à base de nos modèles d'évaluation fondés sur les traces d'apprentissage. Pour ce faire, nous avons modélisé notre activité d'apprentissage à base de situation par une ontologie d'adaptation du domaine de formation. Cette dernière nous a permis une définition de notre base de connaissance en axiomes logiques d'ordre 1 qui permettent d'exprimer des règles existentielles ontologiques. Ces règles nous facilitent le processus de raisonnement sur l'ontologie. Ce processus est décrit par un algorithme que nous avons proposé et qui assure deux types d'adaptation : (i) adaptation de la situation et (ii) adaptation des ressources dans une situation. Notre modèle d'adaptation génère automatiquement des chemins adaptés en utilisant un modèle d'état-transition. Les états sont des situations qui utilisent des ressources. Les transitions sont des étapes permettant de passer d'une situation à une autre en prenant en considération les règles logiques et le mécanisme de raisonnement.

## 7.2 Apports expérimentaux

Pour l'évaluation et la validation de nos contributions sur l'évaluation de l'apprenant et l'adaptation de la ressource et/ou de la situation d'apprentissage, nous avons effectué sur quatre semaines une expérimentation sur des apprenants en Licence Informatique (L2). Ces derniers ont suivi un scénario d'apprentissage composé de quatre situations, à savoir : Cours, TD, TP et Test et qui utilisent des ressources pédagogiques de différentes difficultés. Les traces obtenues durant les sessions d'apprentissage ont été utiles pour l'alimentation de nos modèles d'évaluation et sont utilisées comme des paramètres d'entrée à notre système d'adaptation de la ressource et/ou de la situation d'apprentissage.

## 7.3 Apports à l'éducation numérique

Nos contributions théoriques ont été utilisées comme élément de référence dans le cadre du projet LINUM<sup>1</sup> [Perrin-Doucey et Acerra, 2017a,b] sur la théorisation d'évaluation proposée qui prend en considération la possibilité de rattacher la durée et le nombre de tentatives de l'apprenant à une activité de formation interactive, le but ultime étant une adaptation véritable

<sup>1</sup><http://recherche.gobelins.fr/projets/linum/>

des stratégies de formation et des ressources à chaque sujet, selon ses résultats et ses besoins. En conséquent, nos contributions représentent une brique fondamentale aux attentes de l'éducation sur le numérique, à la fois en termes d'évaluations traçables et de différenciations basées sur des nécessités spécifiques.

## 7.4 Limites

Les recherches que nous avons menées dans cette thèse ont certaines limites et inconvénients. Ci-dessous, nous énumérons les plus importantes identifiées dans nos modèles d'évaluation et dans notre système d'adaptation à base de traces.

### 7.4.1 Limites autour de nos propositions

#### 7.4.1.1 Limites de nos modèles d'évaluation de l'apprenant

- L'inconvénient principal de nos modèles d'évaluation à base de traces consiste dans la difficulté de calculer les paramètres d'atténuation car ils sont fixés par un formateur ou un expert de formation.
- Par ailleurs, évaluer ces paramètres nécessite d'avoir une expertise externe dans le domaine pour prévoir la possibilité de régler ces paramètres en fonction des apprenants.

#### 7.4.1.2 Limites de notre système d'adaptation de la ressource et la stratégie d'apprentissage

- La méthode d'adaptation nécessite une construction et un maintien de la base de connaissances (ontologie) qui doit être validée par les experts du domaine.
- L'algorithme de raisonnement nécessite d'interagir avec une base de données de traces et une ontologie, ce qui impose une mise à jour permanente de ces dernières. Une étape qui reste à implémenter au-delà de la preuve de concept.

## 7.5 Perspectives

Les limites révèlent des perspectives pour les travaux futurs.

À long terme, les perspectives concernent la poursuite de nos travaux de recherche dans le domaine de l'évaluation de l'apprenant et de l'adaptation pédagogique dans d'autres contextes d'application. Dans ce qui suit, nous décrivons quelques pistes intéressantes à nos travaux de recherche.

### **7.5.1 Situation d'apprentissage composée**

Nous proposons le test et la simulation de notre processus d'évaluation de l'apprenant et le processus d'adaptation sur une situation composée d'une ou de plusieurs situations élémentaires. Le test sur ce type complexe de situation peut révéler d'autres paramètres cachés de la situation d'apprentissage à prendre en considération. Cela nécessite un contrôle de cohérence et de vérification des pré-conditions et post-conditions à chaque exécution d'une situation composée.

### **7.5.2 Apprentissage collaboratif**

Dans cette thèse, nous avons proposé l'évaluation et l'adaptation individualisée suivant la performance d'acquisition de l'apprenant durant une situation d'apprentissage. Dans le cas d'un travail collaboratif ou un travail de groupe, comment évaluons-nous : en groupe ou individuellement ? comment adaptons-nous les ressources et les situations ? notre système d'évaluation et d'adaptation doit prendre en considération d'autres traces d'apprentissage comme le nombre d'apprenants impliqués dans une situation d'apprentissage, le temps de passivité ou d'activité de chaque apprenant, le niveau de connaissances de chaque apprenant, etc.

### **7.5.3 Pré-construction du profil apprenant**

Avant l'exécution de l'application, notre système nécessite un profil d'apprenant déjà construit ou un certain nombre de sessions pour qu'il puisse lui attribuer une situation d'apprentissage adaptée suivant son niveau actuel. Nous pouvons améliorer les performances de notre système en proposant l'établissement d'un pré-test et une pré-évaluation de l'apprenant suivant ses objectifs au début de l'interaction avec système pour construire son profil et adapter les situations en conséquence car une meilleure identification de profil permettra une meilleure performance.

### **7.5.4 Distance sémantique**

Une combinaison des outils d'UTL (Usage Tracking Language) avec les traces permet d'avoir une couche sémantique pour les observations des événements, interactions et actions. Avec des règles d'analyses de cette couche, nous pouvons détecter les profils similaires. Dans ce cas, la même logique de pilotage est attribuée aux données d'observations qui sont sémantiquement similaire.

### 7.5.5 Explication et assistance

Un processus d'explication associé à notre processus de notation et d'adaptation pourrait être utile à l'apprenant pour connaître exactement ses faiblesses ; en se focalisant dessus, il peut améliorer son niveau de connaissance par l'exécution de la ressource ou la situation adaptée.

## 7.6 Publications

Nos travaux de recherches ont produit les publications scientifiques internationales suivantes :

### 1. Contributions à l'évaluation des plateformes d'e-Learning :

- Conférence internationale : CSEDU 2016 (International Conference on Computer Supported Education, 2016, Rome, Italy) [Chachoua *et al.*, 2016b] : cet article présente une approche d'évaluation des plateformes d'e-Learning. Après révision, l'article a été sélectionné parmi les meilleurs papiers et une extension de chapitre de livre a été aussi publiée.
- Chapitre de livre : CCIS 2017, volume 739 (Communications in Computer and Information Science) [Chachoua et Tamani, 2016] : ce chapitre présente une approche flexible et hybride d'évaluation des plateformes d'e-Learning.

### 2. Contribution à l'évaluation de l'apprenant fondée sur les traces d'apprentissage :

- Conférence internationale : CSEDU 2016 (International Conference on Computer Supported Education, 2016, Rome, Italy) [Chachoua *et al.*, 2016d] : article court présente nos modèles d'évaluation à base des traces pour l'acquisition des connaissances et l'adaptation des ressources de formation.

### 3. Contribution à l'adaptation de la ressource et la stratégie d'apprentissage à base des traces d'apprentissage :

- Conférence internationale : ICCE 2016 (International Conference on Computers in Education, 2016, IIT Bombay, India) [Chachoua *et al.*, 2016c] : article court présente notre système d'adaptation fondé sur nos modèles d'évaluation à base des traces d'apprentissage. L'article a été sélectionné parmi les meilleurs papiers d'étudiants.

### 4. Contribution sur les systèmes d'apprentissage POLARISQL :

- 
- Conférence internationale : ITS 2016 (Intelligent Tutoring Systems : 13th International Conference, ITS 2016, Zagreb, Croatia) [Chachoua *et al.*, 2016a] : article court (retenu comme poster).



# Bibliographies

- Training and learning architecture (tla) : Experience api (xapi), 2014. URL <https://www.adlnet.gov/adl-research/performance-tracking-analysis/experience-api/>.
- DL ACM : Association for computing machinery (acm) digital library. *Retrieved October, 6:2005*, 2005.
- Adetokunbo Abayomi A ADENOWO et Basirat A ADENOWO : Intelligent tutoring system authoring tools : the design characteristics. *International Journal of Technology Enhanced Learning*, 8(2):114–128, 2016.
- B ADUNA et AL SIRMA : Chapter 6. the serql query language (revision 1.2). user guide for sesame (2005) online resource.
- Mario AMELUNG, Katrin KRIEGER et Daniel ROSNER : E-assessment as a service. *Learning Technologies, IEEE Transactions on*, 4(2):162–174, 2011.
- Mario AMELUNG, Michael PIOTROWSKI et Dietmar RÖSNER : *EduComponents : Experiences in e-assessment in computer science education*, volume 38. ACM, 2006.
- Mario AMELUNG, Michael PIOTROWSKI et Dietmar RÖSNER : educomponents : a component-based e-learning environment. *ACM SIGCSE Bulletin*, 39(3):352–352, 2007.
- John R ANDERSON, Albert T CORBETT, Kenneth R KOEDINGER et Ray PELLETIER : Cognitive tutors : Lessons learned. *The journal of the learning sciences*, 4(2):167–207, 1995.
- John R ANDERSON et Christian J LEBIERE : *The atomic components of thought*. Psychology Press, 2014.
- James H ANDREWS : Testing using log file analysis : tools, methods, and issues. *In Automated Software Engineering, 1998. Proceedings. 13th IEEE International Conference on*, pages 157–166. IEEE, 1998.
- Ivon ARROYO et Beverly Park WOOLF : Inferring learning and attitudes from a bayesian network of log file data. *In AIED*, pages 33–40, 2005.

- Yigal ATTALI : Reliability-based feature weighting for automated essay scoring. *Applied Psychological Measurement*, 39(4):303–313, 2015.
- Walailak ATTHIRAWONG et Bart MACCARTHY : An application of the analytical hierarchy process to international location decision-making. In *Gregory, Mike, Proceedings of The 7th Annual Cambridge International Manufacturing Symposium : Restructuring Global Manufacturing, Cambridge, England : University of Cambridge*, pages 1–18, 2002.
- Peter C AUSTIN : An introduction to propensity score methods for reducing the effects of confounding in observational studies. *Multivariate behavioral research*, 46(3):399–424, 2011.
- Ryan Shaun BAKER, Albert T CORBETT et Kenneth R KOEDINGER : Detecting student misuse of intelligent tutoring systems. In *International conference on intelligent tutoring systems*, pages 531–540. Springer, 2004.
- Ryan Shaun BAKER et Paul Salvador INVENTADO : Educational data mining and learning analytics. In *Learning analytics*, pages 61–75. Springer, 2014.
- Dušan BARAC, Zorica BOGDANOVIĆ, Aleksandar MILIĆ, Branislav JOVANIĆ et Božidar RADENKOVIĆ : Developing adaptive e-learning portal in higher education. In *Toulon-Verona Conference Excellence in Services*, 2015.
- Joseph E BECK et Yue GONG : Wheel-spinning : Students who fail to master a skill. In *International Conference on Artificial Intelligence in Education*, pages 431–440. Springer, 2013.
- Yoav BERGNER, Stefan DROSCHLER, Gerd KORTEMAYER, Saif RAYYAN, Daniel SEATON et David E PRITCHARD : Model-based collaborative filtering analysis of student response data : Machine-learning item response theory. *International Educational Data Mining Society*, 2012.
- Nabila BERKANI et Ladjel BELLATRECHE : A variety-sensitive etl processes. In *International Conference on Database and Expert Systems Applications*, pages 201–216. Springer, 2017.
- Emmanuelle BERNARDIN : Efficacité de l'e-learning en contexte d'entreprise : le rôle de l'environnement d'apprentissage. *Systèmes d'Information et Management*, 12(4):3, 2007.
- John B BIGGS : *Teaching for quality learning at university : What the student does*. McGraw-Hill Education (UK), 2011.
- Robert J BLAKE : The use of technology for second language distance learning. *The Modern Language Journal*, 93(s1):822–835, 2009.

- TAXONOMY MADE EASY BLOOM'S : *Bloom's taxonomy of educational objectives*. Longman, 1965.
- Carol BOSTON : The concept of formative assessment. eric digest. 2002.
- Kevin J BOUDREAU et Karim R LAKHANI : Using the crowd as an innovation partner. *Harvard business review*, 91(4):60–9, 2013.
- Ilyes BOUKHARI, Stéphane JEAN, Idir AIT-SADOUNE et Ladjel BELLATRECHE : The role of user requirements in data repository design. *International journal on software tools for technology transfer*, 20(1):19–34, 2018.
- Colleen BRENT, Andrew WILCOX, Margaret MACDONELL et W Mark NUTT : Web-based learning as a tool for public education and outreach. Rapport technique, Argonne National Lab.(ANL), Argonne, IL (United States), 2017.
- Chris BREW et Claudia LEACOCK : Automated short answer scoring. *Handbook of automated essay evaluation : Current applications and new directions*, 136, 2013.
- Philippe BRIM et Michel BEAUDOUIN-LAFON : A taxonomy and evaluation of formalisms for the specification of interactive systems. In *People and computers X : Proceedings of the HCI'95 conference*, page 197. Cambridge University Press, 1995.
- Christopher BROOKS, Omar CHAVEZ, Jared TRITZ et Stephanie TEASLEY : Reducing selection bias in quasi-experimental educational studies. In *Proceedings of the fifth international conference on learning analytics and knowledge*, pages 295–299. ACM, 2015.
- JM BRUCE et AW BRUCE : Distance education : a learner-centered paradigm. *Clinical laboratory science : journal of the American Society for Medical Technology*, 7(3):178–182, 1993.
- Peter BRUSILOVSKY : Adaptive hypermedia : From intelligent tutoring systems to web-based education. In *Intelligent tutoring systems*, volume 2000, pages 1–7. Springer, 2000.
- Peter BRUSILOVSKY : Adaptive hypermedia. *Modeling and User-Adapted Interaction*, 11:87–110, 2001.
- Peter BRUSILOVSKY : Adaptive hypermedia for education and training. *Adaptive technologies for training and education*, 46, 2012.
- Peter BRUSILOVSKY *et al.* : Adaptive and intelligent technologies for web-based education. *Ki*, 13(4):19–25, 1999.
- Peter BRUSILOVSKY, John EKLUND et Elmar SCHWARZ : Web-based education for all : a tool for development adaptive courseware. *Computer Networks and ISDN Systems*, 30(1-7):291–300, 1998.

- Peter BRUSILOVSKY et Christoph PEYLO : Adaptive and intelligent web-based educational systems. *International Journal of Artificial Intelligence in Education (IJAIED)*, 13:159–172, 2003.
- Daniel BURGOS, Colin TATTERSALL et Rob KOPER : How to represent adaptation in e-learning with ims learning design. *Interactive Learning Environments*, 15(2):161–170, 2007.
- Jill BURSTEIN, Martin CHODOROW et Claudia LEACOCK : Criterionsm online essay evaluation : An application for automated evaluation of student essays. *In IAAI*, pages 3–10, 2003.
- Jill BURSTEIN, Claudia LEACOCK et Richard SWARTZ : Automated evaluation of essays and short answers. 2001.
- Andrew C BUTLER, Jeffrey D KARPICKE et Henry L ROEDIGER III : Correcting a metacognitive error : feedback increases retention of low-confidence correct responses. *Journal of Experimental Psychology : Learning, Memory, and Cognition*, 34(4):918, 2008.
- Brian BUTTERWORTH : *Dyscalculia screener*. NferNelson Pub., 2003.
- Diego CALVANESE, Giuseppe DE GIACOMO, Domenico LEMBO, Maurizio LENZERINI et Riccardo ROSATI : Tractable reasoning and efficient query answering in description logics : The dl-lite family. *Journal of Automated reasoning*, 39(3):385–429, 2007.
- Marie-Françoise CANUT, Sirinya ON-AT, André PÉNINOU et Florence SÈDES : Enrichissement du profil utilisateur à partir de son réseau social dans un contexte dynamique : application d'une méthode de pondération temporelle. 2015.
- Marie-Françoise CANUT, Sirinya ON-AT, André PÉNINOU et Florence SÈDES : Construction du profil social de l'utilisateur dans un contexte dynamique : application d'une méthode de pondération temporelle. *Ingénierie des Systèmes d'Information*, 21(2):65–94, 2016.
- Jaime R CARBONELL : Ai in cai : An artificial-intelligence approach to computer-assisted instruction. *IEEE transactions on man-machine systems*, 11(4):190–202, 1970.
- Zulma CATALDI et Fernando Javier LAGE : Modelado del estudiante en sistemas tutores inteligentes. *TE & ET*, 2010.
- Soraya CHACHOUA, Jamal MALKI et Pascal ESTRILIER : Polarisql : An online tutoring system for learning sql language. *In Intelligent Tutoring Systems : 13th International Conference, ITS 2016, Zagreb, Croatia, June 7-10, 2016. Proceedings*, volume 9684, page 467. Springer, 2016a.

- Soraya CHACHOUA et Nouredine TAMANI : Hybrid ranking method for e-learning platform selection : A flexible approach. *In International Conference on Computer Supported Education*, pages 214–236. Springer, 2016.
- Soraya CHACHOUA, Nouredine TAMANI, Jamal MALKI et Pascal ESTRAILLIER : e-learning platform ranking method using a symbolic approach based on preference relations. *In Proceedings of the 8th International Conference on Computer Supported Education*, pages 114–122, 2016b. ISBN 978-989-758-179-3.
- Soraya CHACHOUA, Nouredine TAMANI, Jamal MALKI et Pascal ESTRAILLIER : Towards a trace-based adaptation model in e-learning systems. *In Proceedings of the 24th International Conference on Computers in Education. India : Asia-Pacific Society for Computers in Education, ICCE 2016*, 2016c.
- Soraya CHACHOUA, Nouredine TAMANI, Jamal MALKI et Pascal ESTRAILLIER : Towards a trace-based evaluation model for knowledge acquisition and training resource adaption. *In Proceedings of the 8th International Conference on Computer Supported Education*, pages 121–128, 2016d. ISBN 978-989-758-179-3.
- Hsuan-Pu CHANG, Chun-Chia WANG, Kuen Han JAN et Timothy K SHIH : Scorm sequencing testing for sequencing control mode. *In Advanced Information Networking and Applications, 2006. AINA 2006. 20th International Conference on*, volume 2, pages 5–pp. IEEE, 2006.
- Suming CHEN, Arthur CHOI et Adnan DARWICHE : Computer adaptive testing using the same-decision probability. *In Proceedings of the Twelfth UAI Conference on Bayesian Modeling Applications Workshop-Volume 1565*, pages 34–43. CEUR-WS. org, 2015.
- Murray CHUCK, Inseok Chong EUGENE, Das SOURI, Kolovski VLADIMIR, Perry MATT, Rieb KARL, Srinivasan JAGS, Sundara SEEMA, (Alan) Wu ZHE et Yalamanchi ARAVIND : Oracle database semantic technologies developer's guide, 11g release 2 (11.2). 2013. URL [https://docs.oracle.com/cd/E11882\\_01/appdev.112/e25609/title.htm](https://docs.oracle.com/cd/E11882_01/appdev.112/e25609/title.htm).
- Liviu CIORTEA, Cristian ZAMFIR, Stefan BUCUR, Vitaly CHIPOUNOV et George CANDEA : Cloud9 : A software testing service. *ACM SIGOPS Operating Systems Review*, 43(4):5–10, 2010.
- Cheryl A CISERO, James M ROYER, Horace G MARCHANT III et Stanley J JACKSON : Can the computer-based academic assessment system (caas) be used to diagnose reading disability in college students ? *Journal of Educational Psychology*, 89(4):599, 1997.
- Jacob COHEN : *Statistical power analysis for the behavioral sciences (rev.* Lawrence Erlbaum Associates, Inc, 1977.

- Cristina CONATI, Abigail GERTNER et Kurt VANLEHN : Using bayesian networks to manage uncertainty in student modeling. *User modeling and user-adapted interaction*, 12(4):371–417, 2002.
- Cristina CONATI et Xiaohong ZHAO : Building and evaluating an intelligent pedagogical agent to improve the effectiveness of an educational game. *In Proceedings of the 9th international conference on Intelligent user interfaces*, pages 6–13. ACM, 2004.
- William CONDON : Large-scale assessment, locally-developed measures, and automated scoring of essays : Fishing for red herrings? *Assessing Writing*, 18(1):100–108, 2013.
- Ricardo CONEJO, Eva MILLÁN, José Luis Pérez de la CRUZ et Mónica TRELLA : Modelado del alumno : un enfoque bayesiano. *Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial*, 5(12), 2001.
- Geoffrey CRISP : Interactive e-assessment : moving beyond multiple-choice questions. *Centre for Learning and Professional Development. Adelaide : University of Adelaide*, 3:12–31, 2009.
- Terence J CROOKS : The impact of classroom evaluation practices on students. *Review of educational research*, 58(4):438–481, 1988.
- Souripriya DAS et Jagannathan SRINIVASAN : Database technologies for rdf. *In Reasoning Web International Summer School*, pages 205–221. Springer, 2009.
- Jovan DAŠIĆ, Predrag DAŠIĆ et Veis ŠERIFI : Evolution of e-learning. *Learning*, page 4, 2010.
- P DAŠIĆ, V NEDEFF et S ŠEHOVIĆ : Adaptive hypermedia systems in e-learning. *Journal of Modelling and Optimization in the Machines Building Fields (MOCM)*, 13(3):384–395, 2007.
- Predrag DAŠIĆ, Jovan DAŠIĆ, Bojan CRVENKOVIĆ et Veis ŠERIFI : A review of intelligent tutoring systems in e-learning. 2016.
- Semire DIKLI : An overview of automated scoring of essays. *The Journal of Technology, Learning and Assessment*, 5(1), 2006.
- Tarek DJOUAD, Lotfi Sofiane SETTOUTI, Yannick PRIÉ, Christophe REFFAY et Alain MILLE : Un système à base de traces pour la modélisation et l’élaboration d’indicateurs d’activités éducatives individuelles et collectives. mise à l’épreuve sur moodle. *Mise à l’épreuve sur Moodle. Technique et Science Informatiques*, 2010.
- John DUNLOSKY, Katherine A RAWSON, Elizabeth J MARSH, Mitchell J NATHAN et Daniel T WILLINGHAM : Improving students’ learning with effective learning techniques : Promising

- directions from cognitive and educational psychology. *Psychological Science in the Public Interest*, 14(1):4–58, 2013.
- Wenger ETIENNE : Artificial intelligence and tutoring systems, 1987.
- Richard M FELDER et Linda K SILVERMAN : Learning and teaching styles in engineering education. *Engineering education*, 78(7):674–681, 1988.
- Richard M FELDER et Joni SPURLIN : Applications, reliability and validity of the index of learning styles. *International journal of engineering education*, 21(1):103–112, 2005.
- William FODDY : *Constructing questions for interviews and questionnaires : Theory and practice in social research*. Cambridge university press, 1994.
- Rick GARLIKOV : The socratic method : Teaching by asking instead of by telling. *Website*, [http://www.garlikov.com/Soc\\_Meth.html](http://www.garlikov.com/Soc_Meth.html), 2001.
- Dragan GAŠEVIĆ et Marek HATALA : Ontology mappings to improve learning resource search. *British Journal of Educational Technology*, 37(3):375–389, 2006.
- Joyce Wangui GIKANDI, Donna MORROW et Niki E DAVIS : Online formative assessment in higher education : A review of the literature. *Computers & Education*, 57(4):2333–2351, 2011.
- Wanda E GILL : The ready to teach program : A federal initiative in support of online courses for teachers. *Online Submission*, 2011.
- Michael GLASS : Processing language input in the circsim-tutor intelligent tutoring system. *In Artificial Intelligence in Education*, pages 210–221, 2001.
- Stefan GÖBEL, Florian MEHM, Sabrina RADKE et Ralf STEINMETZ : 80days : Adaptive digital storytelling for digital educational games. *In Proceedings of the 2nd International Workshop on Story-Telling and Educational Games (STEG'09)*, volume 498, 2009.
- Paulo GOMES, Bruno ANTUNES, Luis RODRIGUES, Arnaldo SANTOS, Jacinto BARBEIRA et Rafael CARVALHO : Using ontologies for elearning personalization. *Communication & Cognition*, 41(1):127, 2008.
- John T GORGONE, Gordon B DAVIS, Joseph S VALACICH, Heikki TOPI, David L FEINSTEIN et Herbert E LONGENECKER JR : Model curriculum and guidelines for undergraduate degree programs in information systems. *Association for Computing Machinery (ACM), Association for Information Systems (AIS), Association of Information Technology Professionals (AITP)*, 2002.

- Arthur C GRAESSER, Shulan LU, George Tanner JACKSON, Heather Hite MITCHELL, Mathew VENTURA, Andrew OLNEY et Max M LOUWERSE : Autotutor : A tutor with dialogue in natural language. *Behavior Research Methods*, 36(2):180–192, 2004.
- EA GRAF et JH FIFE : Difficulty modeling and automatic generation of quantitative items : recent advances and possible next steps. *Automatic Item Generation : Theory and Practice*, pages 157–179, 2012.
- Sabine GRAF : *Adaptivity in learning management systems focussing on learning styles*. Thèse de doctorat, Vienna University of Technology, 2007.
- Sabine GRAF et Beate LIST : An evaluation of open source e-learning platforms stressing adaptation issues. *In Proceedings of the 5th IEEE International Conference on Advanced Learning Technologies, ICALT'05*, pages 163–165. IEEE Computer press, 2005.
- Jane GREENBERG et W Davenport ROBERTSON : Semantic web construction : an inquiry of authors' views on collaborative metadata generation. *In International Conference on Dublin Core and Metadata Applications*, pages 45–52, 2002.
- Jim E GREER et Gordon I MCCALLA : *Student modelling : the key to individualized knowledge-based instruction*, volume 125. Springer Science & Business Media, 2013.
- Foteini GRIVOKOSTOPOULOU et Ioannis HATZILYGEROUDIS : Teaching ai search algorithms in a web-based educational system. *International Association for Development of the Information Society*, 2013.
- Foteini GRIVOKOSTOPOULOU, Isidoros PERIKOS et Ioannis HATZILYGEROUDIS : An educational system for learning search algorithms and automatically assessing student performance. *International Journal of Artificial Intelligence in Education*, 27(1):207–240, 2017.
- Begoña GROS, Marcelo MAINA *et al.* : *The Future of Ubiquitous Learning : Learning Designs for Emerging Pedagogies*. Springer, 2015.
- Philip J. GUO : Online Python Tutor : Embeddable web-based program visualization for CS education. *In Proceedings of the 44th ACM Technical Symposium on Computer Science Education, SIGCSE'13*, pages 579–584, New York, NY, USA, 2013a. ACM. ISBN 978-1-4503-1868-6. URL <http://doi.acm.org/10.1145/2445196.2445368>.
- Philip J GUO : Online python tutor : embeddable web-based program visualization for cs education. *In Proceeding of the 44th ACM technical symposium on Computer science education*, pages 579–584. ACM, 2013b. URL <http://pythontutor.com/>.
- C. HADJI : *L'évaluation à l'école*. Numéro 9782091222707 in Questions d'enseignants. 2015 édition, Août 2015.



- Thair M HAMTINI et Hussam Nawwaf FAKHOURI : Evaluation of open-source e-learning platforms based on the qualitative weight and sum approach and analytic hierarchy process. Rapport technique, Retrieved 2013/05/21 from [http://www.iiis.org/CDs2012/CD2012SCI/IMSCI\\_2012/PapersPdf/EA418WG.pdf](http://www.iiis.org/CDs2012/CD2012SCI/IMSCI_2012/PapersPdf/EA418WG.pdf), 2012.
- Manoj Chandra HANDA *et al.* : Learner-centred differentiation model : A new framework. *Australasian Journal of Gifted Education*, 18(2):55, 2009.
- Jiangang HAO, Zhan SHU et Alina von DAVIER : Analyzing process data from game/scenario-based tasks : an edit distance approach. *JEDM-Journal of Educational Data Mining*, 7(1):33–50, 2015.
- Steve HARRIS, Andy SEABORNE et Eric PRUD'HOMMEAUX : Sparql 1.1 query language. *W3C recommendation*, 21(10), 2013.
- Michael HRUSKA, R LONG, C AMBURN, Tara KILCULLEN et T POEPELMAN : Experience api and team evaluation : evolving interoperable performance assessment. *In The Interservice/Industry Training, Simulation & Education Conference (IITSEC)*, 2014.
- Carlos HUERTAS et Reyes JUÁREZ-RAMÍREZ : Developing an intelligent tutoring system for vehicle dynamics. *Procedia-Social and Behavioral Sciences*, 106:838–847, 2013.
- Gwo-Jen HWANG, Fan-Ray KUO, Peng-Yeng YIN et Kuo-Hsien CHUANG : A heuristic algorithm for planning personalized learning paths for context-aware ubiquitous learning. *Computers & Education*, 54(2):404–415, 2010.
- E JOSEPH : Engagement tracing : using response times to model student disengagement. *Artificial intelligence in education : Supporting learning through intelligent and socially informed technology*, 125:88, 2005.
- Gregory KARVOUNARAKIS, Aimilia MAGKANARAKI, Sophia ALEXAKI, Vassilis CHRISTOPHIDES, Dimitris PLEXOUSAKIS, Michel SCHOLL et Karsten TOLLE : Rql : A functional query language for rdf. *In The Functional Approach to Data Management*, pages 435–465. Springer, 2004.
- Tanja KÄSER, Gian-Marco BASCHERA, Juliane KOHN, Karin KUCIAN, Verena RICHTMANN, Ursina GROND, Markus GROSS et Michael von ASTER : Design and evaluation of the computer-based training program calcularis for enhancing numerical cognition. *Frontiers in psychology*, 4, 2013a.
- Tanja KÄSER, Alberto Giovanni Busetto, Barbara SOLENTHALER, Juliane KOHN, Michael von ASTER et Markus GROSS : Cluster-based prediction of mathematical learning patterns. *In International Conference on Artificial Intelligence in Education*, pages 389–399. Springer, 2013b.

- Spiro KIOUSIS : Interactivity : a concept explication. *New media & society*, 4(3):355–383, 2002.
- Aniket KITTUR, Ed H CHI et Bongwon SUH : Crowdsourcing user studies with mechanical turk. *In Proceedings of the SIGCHI conference on human factors in computing systems*, pages 453–456. ACM, 2008.
- Severin KLINGLER, Tanja KÄSER, Alberto-Giovanni Busetto, Barbara SOLENTHALER, Juliane KOHN, Michael von ASTER et Markus GROSS : Stealth assessment in its-a study for developmental dyscalculia. *In International Conference on Intelligent Tutoring Systems*, pages 79–89. Springer, 2016.
- Evgeny KNUTOV, Paul DE BRA et Mykola PECHENIZKIY : Ah 12 years later : a comprehensive survey of adaptive hypermedia methods and techniques. *New Review of Hypermedia and Multimedia*, 15(1):5–38, 2009.
- Viswanathan KODAGANALLUR, Rob R WEITZ et David ROSENTHAL : A comparison of model-tracing and constraint-based intelligent tutoring paradigms. *International Journal of Artificial Intelligence in Education*, 15(2):117–144, 2005.
- Ken KOEDINGER et Michael TANNER : 7 things you should know about intelligent tutoring systems. Rapport technique, EDUCAUSE, 2013.
- RB KOZMA : Transforming education : Assessing and teaching 21st century skills. *The transition to computer-based assessment*, 13(21):207, 2009.
- Rohit KUMAR, Gregory KWK CHUNG, Ayesha MADNI et Bruce ROBERTS : First evaluation of the physics instantiation of a problem-solving-based online learning platform. *In Artificial Intelligence in Education*, pages 686–689. Springer, 2015.
- Julien LAFLAQUIERE, Lotfi S SETTOUTI, Yannick PRIÉ et Alain MILLE : Trace-based framework for experience management and engineering. *In Knowledge-Based Intelligent Information and Engineering Systems*, pages 1171–1178. Springer, 2006.
- Andrew S LAN, Andrew E WATERS, Christoph STUDER et Richard G BARANIUK : Sparse factor analysis for learning and content analytics. *Journal of Machine Learning Research*, 15(1):1959–2008, 2014.
- Thomas K LANDAUER, Peter W FOLTZ et Darrell LAHAM : An introduction to latent semantic analysis. *Discourse processes*, 25(2-3):259–284, 1998.
- Thomas K LANDAUER, Darrell LAHAM et Peter FOLTZ : Automatic essay assessment. assessment in e-ducation : Principles. *Policy and Practice*, 10(3):295–309, 2003.

- Diana LAURILLARD : *Rethinking university teaching : A conversational framework for the effective use of learning technologies*. Routledge, 2013.
- Claudia LEACOCK et Martin CHODOROW : C-rater : Automated scoring of short-answer questions. *Computers and the Humanities*, 37(4):389–405, 2003.
- Longzhuang LI, Hongchi SHI, Yi SHANG et Su-Shing CHEN : Open learning objects for data structure course. *Journal of Computing Sciences in Colleges*, 18(4):56–64, 2003.
- Yunyao LI, Rajasekar KRISHNAMURTHY, Sriram RAGHAVAN, Shivakumar VAITHYANATHAN et HV JAGADISH : Regular expression learning for information extraction. *In Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 21–30. Association for Computational Linguistics, 2008. URL <https://regex101.com/r/gH2jE1/1>.
- Zhi LIU et Haiquan WANG : A modeling method based on bayesian networks in intelligent tutoring system. *In Computer Supported Cooperative Work in Design, 2007. CSCWD 2007. 11th International Conference on*, pages 967–972. IEEE, 2007.
- Markus LUCKEY et Gregor ENGELS : High-quality specification of self-adaptive software systems. *In Software Engineering for Adaptive and Self-Managing Systems (SEAMS), 2013 ICSE Workshop on*, pages 143–152. IEEE, 2013.
- Danny LYNCH et Colm P HOWLIN : Real world usage of an adaptive testing algorithm to uncover latent knowledge. *In 7th international conference of education, research and innovation (ICERI2014 proceedings). IATED, Seville, Spain*, pages 504–511, 2014.
- Urszula MARKOWSKA KACZMAR, Halina KWASNICKA et Mariusz PARADOWSKI : Intelligent techniques in personalization of learning in e-learning systems. *In Computational Intelligence for Technology Enhanced Learning*, pages 1–23. Springer, 2010.
- Brent MARTIN, Kenneth R KOEDINGER, Antonija MITROVIC et Santosh MATHAN : On using learning curves to evaluate its. *In AIED*, pages 419–426, 2005.
- Brent MARTIN, Antonija MITROVIC, Kenneth R KOEDINGER et Santosh MATHAN : Evaluating and improving adaptive educational systems with learning curves. *User Modeling and User-Adapted Interaction*, 21(3):249–283, 2011.
- Romain MARTIN : New possibilities and challenges for assessment through the use of technology. *Towards a research agenda on computer-based assessment*, page 6, 2008.
- Noboru MATSUDA, William W COHEN et Kenneth R KOEDINGER : Teaching the teacher : tutoring simstudent leads to more effective cognitive tutor authoring. *International Journal of Artificial Intelligence in Education*, 25(1):1–34, 2015.

- Wafa MEFTEH : *Approche ontologique pour la modélisation et le raisonnement sur les trajectoires : prise en compte des aspects thématiques, temporels et spatiaux. (Ontological approach for modeling and reasoning about trajectories : taking into account the thematic, temporal and spatial aspects)*. Thèse de doctorat, University of La Rochelle, France, 2013.  
URL <https://tel.archives-ouvertes.fr/tel-01124314>.
- Janet METCALFE, Nate KORNELL et Bridgid FINN : Delayed versus immediate feedback in children's and adults' vocabulary learning. *Memory & Cognition*, 37(8):1077–1087, 2009.
- Alistair MILES et Dan BRICKLEY : Skos core vocabulary specification. Retrieved June, 14: 2008, 2005.
- Eva MILLÁN, Tomasz LOBODA et Jose Luis Pérez-de-la CRUZ : Bayesian networks for student model engineering. *Computers & Education*, 55(4):1663–1683, 2010.
- Alain MILLE, Pierre-Antoine CHAMPIN, Amélie CORDIER, Olivier GEORGEON et Marie LEFEVRE : Trace-based reasoning-modeling interaction traces for reasoning on experiences. *In The 26th International FLAIRS Conference*, pages 1–15, 2013.
- Through Educational Data MINING : Enhancing teaching and learning through educational data mining and learning analytics : An issue brief. *In Proceedings of conference on advanced technology for education*, 2012.
- Ayse Tosun MISIRLI et Ayse Basar BENER : Bayesian networks for evidence-based decision-making in software engineering. *IEEE Transactions on Software Engineering*, 40(6):533–554, 2014.
- Robert J MISLEVY, John T BEHRENS, Kristen E DICERBO et Roy LEVY : Design and discovery in educational assessment : Evidence-centered design, psychometrics, and educational data mining. *JEDM-Journal of Educational Data Mining*, 4(1):11–48, 2012.
- Antonija MITROVIC, Brent MARTIN et Pramuditha SURAWEERA : Intelligent tutors for all : The constraint-based approach. *IEEE Intelligent Systems*, (4):38–45, 2007.
- Edna H MORY : Feedback research revisited. *Handbook of research on educational communications and technology*, 2:745–783, 2004.
- Tom MURRAY : Formative qualitative evaluation for " exploratory " its research. *Journal of Interactive Learning Research*, 4(2):179, 1993.
- Tom MURRAY : Authoring intelligent tutoring systems : An analysis of the state of the art. *International Journal of Artificial Intelligence in Education (IJAIED)*, 10:98–129, 1999.

- Yasir Eltigani Ali MUSTAFA et Sami Mohamed SHARIF : An approach to adaptive e-learning hypermedia system based on learning styles (AEHS-LS) : Implementation and evaluation. *International Journal of Library and Information Science*, 3(1):15–28, 2011.
- David NICOL : E-assessment by design : using multiple-choice tests to good effect. *Journal of Further and Higher Education*, 31(1):53–64, 2007.
- David J NICOL et Debra MACFARLANE-DICK : Formative assessment and self-regulated learning : A model and seven principles of good feedback practice. *Studies in higher education*, 31(2):199–218, 2006.
- Stellan OHLSSON : Some principles of intelligent tutoring. *In Artificial intelligence and education*, volume 1, pages 203–238, 1987.
- University of Zurich OLAT TEAM (OVERALL) : *OLAT 7.8-Manuel d'utilisation*. 2014.
- Steven OXMAN, William WONG et DVX INNOVATIONS : White paper : Adaptive learning systems. *Integrated Education Solutions*, 2014.
- Rena M PALLOFF et Keith PRATT : *Assessing the online learner : Resources and strategies for faculty*, volume 14. John Wiley & Sons, 2008.
- Zacharoula PAPAMITSIOU et Anastasios A ECONOMIDES : Temporal learning analytics visualizations for increasing awareness during assessment. *RUSC. Universities and Knowledge Society Journal*, 12(3):129–147, 2015.
- Ji-Hye PARK et Hee Jun CHOI : Factors influencing adult learners' decision to drop out or persist in online learning. *Educational Technology & Society*, 12(4):207–217, 2009.
- Ok-choon PARK et Jung LEE : Adaptive instructional systems. *Educational Technology Research and Development*, 25:651–684, 2003.
- Harrie PASSIER et JT JEURING : Ontology based feedback generation in design-orientated e-learning systems. *In Proceedings of the IADIS International Conference, e-Society 2004.*, pages 992–996, 2004.
- Thanos PATELIS : An overview of computer-based testing. 2000.
- Judea PEARL : *Probabilistic reasoning in intelligent systems : networks of plausible inference*. Elsevier, 2014.
- Neil PEIRCE, Owen CONLAN et Vincent WADE : Adaptive educational games : Providing non-invasive personalised learning experiences. *In Digital Games and Intelligent Toys Based Education, 2008 Second IEEE International Conference on*, pages 28–35. IEEE, 2008.

- Alejandro PEÑA-AYALA : Educational data mining : A survey and a data mining-based analysis of recent works. *Expert systems with applications*, 41(4):1432–1462, 2014.
- Les PERELMAN : When “the state of the art” is counting words. *Assessing Writing*, 21:104–111, 2014.
- Agnès PERRIN-DOUCEY et Eleonora ACERRA : Linum : Un projet numérique pour interroger l’enseignement de la lecture littéraire. *Revue de recherches en littératie médiatique multimodale*, (8), 2017a.
- Agnès PERRIN-DOUCEY et Eleonora ACERRA : Penser la lecture littéraire par et pour le numérique : conception, activités et usages en question. *Repères. Recherches en didactique du français langue maternelle*, (56):197–213, 2017b.
- Phuong Thao PHAM : *Architecture à base de situations pour le traitement des quiproquos dans l’exécution adaptative d’applications interactives*. Thèse de doctorat, Université de La Rochelle, 2013.
- Phuong Thao PHAM, Mourad RABAH et Pascal ESTRAILLIER : Handling the misunderstanding in interactions : Definition and solution. *In International Conference on Software Engineering & Applications*, pages 47–52, 2011.
- Phuong Thao PHAM, Mourad RABAH et Pascal ESTRAILLIER : Agent-based architecture and situation-based scenario for consistency management. *In Computer Science and Information Systems (FedCSIS), 2013 Federated Conference on*, pages 1041–1046. IEEE, 2013.
- Pipatsarun PHOBUN et Jiracha VICHEANPANYA : Adaptive intelligent tutoring systems for e-learning systems. *Procedia-Social and Behavioral Sciences*, 2(2):4064–4069, 2010.
- David L POOLE et Alan K MACKWORTH : *Artificial Intelligence : foundations of computational agents*. Cambridge University Press, 2010.
- Eric PRUD, Andy SEABORNE *et al.* : Sparql query language for rdf. 2006.
- Božidar Lj RADENKOVIĆ, Marijana S DESPOTOVIĆ-ZRAKIĆ, Dušan M BARAC, Zorica M BOGDANOVIĆ et Aleksandar R MILIĆ : Web portal for adaptive e-learning. *In Telecommunication in Modern Satellite Cable and Broadcasting Services (TELSIKS), 2011 10th International Conference on*, volume 1, pages 365–368. IEEE, 2011.
- Chaitanya RAMINENI et David M WILLIAMSON : Automated essay scoring : Psychometric guidelines and practices. *Assessing Writing*, 18(1):25–39, 2013.
- Alan RAMÍREZ-NORIEGA, Reyes JUÁREZ-RAMÍREZ et Yobani MARTÍNEZ-RAMÍREZ : Evaluation module based on bayesian networks to intelligent tutoring systems. *International Journal of Information Management*, 37(1):1488–1498, 2017.

- Mohammed A RAZEK et Hisham Jameel A BARDESI : Adaptive course for mobile learning. *In Computational Intelligence, Communication Systems and Networks (CICSyN), 2013 Fifth International Conference on*, pages 328–333. IEEE, 2013.
- Charles M REIGELUTH, Sinem ASLAN, Zengguan CHEN, Pratima DUTTA, Yeol HUH, Dabae LEE, Chun-Yi LIN, Ya-Huei LU, Mina MIN, Verily TAN *et al.* : Personalized integrated educational system : Technology functions for the learner-centered paradigm of education. *Journal of Educational Computing Research*, 53(3):459–496, 2015.
- Marta REY-LÓPEZ, Rebeca P DÍAZ-REDONDO, Ana FERNÁNDEZ-VILAS, José J PAZOS-ARIAS, Jorge GARCÍA-DUQUE, Alberto GIL-SOLLA et Manuel RAMOS-CABRER : An extension to the adl scorm standard to support adaptivity : The t-learning case-study. *Computer Standards & Interfaces*, 31(2):309–318, 2009.
- Elaine RICH : User modeling via stereotypes. *Cognitive science*, 3(4):329–354, 1979.
- Manuel RIVAS NAVARRO : *Procesos cognitivos y aprendizaje significativo*. Comunidad de Madrid. Consejería de Educación. Viceconsejería de Organización Educativa, 2008.
- Ido ROLL, Ryan Shaun BAKER, Vincent ALEVEN et Kenneth R KOEDINGER : A metacognitive act-r model of students' learning strategies in intelligent tutoring systems. *In International Conference on Intelligent Tutoring Systems*, pages 854–856. Springer, 2004.
- Marta C ROSATELLI et John A SELF : A collaborative case study system for distance learning. *International Journal of Artificial Intelligence in Education*, 14(1):97–125, 2004.
- Liam ROURKE, Terry ANDERSON, D Randy GARRISON et Walter ARCHER : Methodological issues in the content analysis of computer conference transcripts. *International journal of artificial intelligence in education (IJAIED)*, 12:8–22, 2001.
- Judith Haymore SANDHOLTZ, Cathy RINGSTAFF et David C Dwyer NEW : The computer enters the classroom. *Teachers college record*, 100(3):656–669, 1999.
- R SANTHI, B PRIYA et JM NANDHINI : Review of intelligent tutoring systems using bayesian approach. *arXiv preprint arXiv :1302.7081*, 2013.
- Friedrich SCHEUERMANN et Julius BJÖRNSSON : The transition to computer-based assessment. *Luxembourg : Office for Official Publications of the European Communities*, 2009.
- SCORM : *SCORM 2004 Handbook*. The e-Learning Consortium, Japan., March 2006. URL <http://203.183.1.152/aen/content/act2005eg/data/txt1.pdf>. Version 1.04.
- Lotfi Sofiane SETTOUTI, Yannick PRIÉ, Pierre-Antoine CHAMPIN, Jean-Charles MARTY et Alain MILLE : A trace-based systems framework : Models, languages and semantics. 2009a.

- Lotfi Sofiane SETTOUTI, Yannick PRIÉ, Damien CRAM, Pierre-Antoine CHAMPIN et Alain MILLE : A trace-based framework for supporting digital object memories. *In Workshops Proceedings of the 5th International Conference on Intelligent Environments, Barcelona, Spain, 19th of July, 2009*, pages 39–44, 2009b.
- Lotfi Sofiane SETTOUTI, Yannick PRIÉ, Jean-Charles MARTY et Alain MILLE : A trace-based system for technology-enhanced learning systems personalisation. *In The 9th IEEE International Conference on Advanced Learning Technologies, ICALT 2009, Riga, Latvia, July 15-17, 2009*, pages 93–97, 2009c. URL <http://dx.doi.org/10.1109/ICALT.2009.159>.
- Marianne SHAW, Landon DETWILER, James F BRINKLEY et Dan SUCIU : A dataflow graph transformation language and query rewriting system for rdf ontologies. *In SSDBM*, pages 544–561. Springer, 2012.
- Valerie J SHUTE : Stealth assessment in computer-based games to support learning. *Computer games and instruction*, 55(2):503–524, 2011.
- Valerie J SHUTE et Diego ZAPATA-RIVERA : Adaptive educational systems. *Adaptive technologies for training and education*, 7:27, 2012.
- Marián ŠIMKO, Michal BARLA et Mária BIELIKOVÁ : Alef : A framework for adaptive web-based learning 2.0. *Key Competencies in the Knowledge Society*, pages 367–378, 2010.
- Stephen SIRECI et Richard M LUECHT : A review of models for computer-based testing. 2012.
- Thomas D SNYDER, Cristobal de BREY et Sally A DILLOW : Digest of education statistics 2014, nces 2016-006. *National Center for Education Statistics*, 2016.
- Sergey SOSNOVSKY et Peter BRUSILOVSKY : Evaluation of topic-based adaptation and student modeling in quizguide. *User Modeling and User-Adapted Interaction*, 25(4):371–424, 2015.
- R SOTTILARE, A GRAESSER, X HU et H HOLDEN : Design recommendations for intelligent tutoring systems : Learner modeling (vol. 1). *Orlando, FL : Army Research Laboratory*, 2013.
- SPHERE RESEARCH LABS : ideone.com - Online Compiler and IDE, 29 décembre 2016. URL <https://ideone.com/>.
- Steven E STEMLER : A comparison of consensus, consistency, and measurement approaches to estimating interrater reliability. *Practical Assessment, Research & Evaluation*, 9(4):1–19, 2004.



- Andrea STERBINI et Marco TEMPERINI : Adaptive construction and delivery of web-based learning paths. *In Frontiers in Education Conference, 2009. FIE'09. 39th IEEE*, pages 1–6. IEEE, 2009.
- Daniel L STUFFLEBEAM : Empowerment evaluation, objectivist evaluation, and evaluation standards : Where the future of evaluation should not go and where it needs to go. *Evaluation practice*, 15(3):321–338, 1994.
- Pramuditha SURAWEERA et Antonija MITROVIC : Kermit : A constraint-based tutor for database modeling. *In International Conference on Intelligent Tutoring Systems*, pages 377–387. Springer, 2002.
- Nguyen THAI-NGHE, Lucas DRUMOND, Tomáš HORVÁTH, Lars SCHMIDT-THIEME *et al.* : Multi-relational factorization models for predicting student performance. *In Proc. of the KDD Workshop on Knowledge Discovery in Educational Data*, pages 27–40, 2011.
- N THOMPSON et D WIESS : Computerised and adaptive testing in educational assessment. *The transition to computer-based assessment. New approaches to skills assessment and implications for large-scale testing*, pages 127–133, 2009.
- Ivelisse Teresa Machín TORRES et Vivian Estrada SENTÍ : Intelligent tutor system for learning object oriented programming. 2017.
- TUTORIALS POINT TEAM : Free online IDE and terminal, 29 décembre 2016. URL <https://www.tutorialspoint.com/index.htm>.
- Wim J van der LINDEN et Cees AW GLAS : *Elements of adaptive testing*. Springer, 2010.
- JR VAN SETERS, MA OSSEVOORT, J TRAMPER et Martin J GOEDHART : The influence of student characteristics on the use of adaptive e-learning material. *Computers & Education*, 58(3):942–952, 2012.
- Mieke VANDEWAETERE, Piet DESMET et Geraldine CLAREBOUT : The contribution of learner characteristics in the development of computer-based adaptive learning environments. *Computers in Human Behavior*, 27(1):118–130, 2011.
- Kurt VANLEHN, Pamela W JORDAN, Carolyn P ROSÉ, Dumisizwe BHEMBE, Michael BÖTTNER, Andy GAYDOS, Maxim MAKATCHEV, Umarani PAPPUSWAMY, Michael RINGENBERG, Antonio ROQUE *et al.* : The architecture of why2-atlas : A coach for qualitative physics essay writing. *In International Conference on Intelligent Tutoring Systems*, pages 158–167. Springer, 2002.
- Anthony J VIERA, Joanne M GARRETT *et al.* : Understanding interobserver agreement : the kappa statistic. *Fam Med*, 37(5):360–363, 2005.

- Michael von ASTER, Monika Weinhold ZULAUF et Ralf HORN : *Neuropsychologische Test-batterie für Zahlenverarbeitung und Rechnen bei Kindern : ZAREKI-R*. Harcourt, 2006.
- Michael G VON ASTER et Ruth S SHALEV : Number development and developmental dyscalculia. *Developmental Medicine & Child Neurology*, 49(11):868–873, 2007.
- Jakob WANDALL : National tests in Denmark—CAT as a pedagogic tool. *Association of Test Publishers*, 12(1), 2011.
- Shu-Lin WANG et Chun-Yi WU : Application of context-aware and personalized recommendation to implement an adaptive ubiquitous learning system. *Expert Systems with Applications*, 38(9):10831–10838, 2011.
- Joe WARREN, Scott RIXNER, John GREINER et Stephen WONG : Facilitating human interaction in an online programming course. *In Proceedings of the 45th ACM technical symposium on Computer science education*, pages 665–670. ACM, 2014. URL <http://www.codeskulptor.org/>.
- Barbara WASSON : Advanced educational technologies : The learning environment. *Computers in human behavior*, 13(4):571–594, 1997.
- Gerhard WEBER et Marcus SPECHT : User modeling and adaptive navigation support in WWW-based tutoring systems. *In User Modeling*, pages 289–300. Springer, 1997.
- Oliver WILHELM : Issues in computerized ability measurement : Getting out of the jingle and jangle jungle. *The transition to computer-based assessment*, pages 145–150, 2009.
- David M WILLIAMSON, Xiaoming XI et F Jay BREYER : A framework for evaluation and use of automated scoring. *Educational measurement : issues and practice*, 31(1):2–13, 2012.
- Yongwei YANG, Chad W BUCKENDAHL, Piotr J JUSZKIEWICZ et Dennison S BHOLA : A review of strategies for validating computer-automated scoring. *Applied Measurement in Education*, 15(4):391–412, 2002.
- Telmo ZARRAONANDÍA, Paloma DÍAZ et Ignacio AEDO : Modeling games for adaptive and personalized learning. *In The Future of Ubiquitous Learning*, pages 217–239. Springer, 2016.
- Jianping ZHANG : Intelligent tutoring systems : research status and its development in China. *In Natural Language Processing and Knowledge Engineering, 2005. IEEE NLP-KE'05. Proceedings of 2005 IEEE International Conference on*, pages 683–689. IEEE, 2005.

---

Abdelhamid ZOUHAIR, Benaïssa AMAMI, Hadhoum BOUKACHOUR, Patrick PERSON, Cyrille BERTELLE *et al.* : Dynamic case-based reasoning based on the multi-agent systems : Individualized follow-up of learners in distance learning. *In Intelligent Distributed Computing VI*, pages 51–57. Springer, 2013.