



APPRIVOISER L'HÉTÉROGÉNÉITÉ EN INFORMATIQUE 1^{ÈRE} ANNÉE

Nadia BENNANI¹, Sylvie CAZALENS¹, Vincent CHEUTET², Claire LESCHI¹, Odysée MERVEILLE³, Camille MORIOT⁴, Delphine MULLER³, Timothée PECATTE¹, Catherine POTHIER¹, Christophe RIGOTTI⁵, Hervé RIVANO⁴ et Nicolas STOULS⁴

¹Univ Lyon, INSA-Lyon, CNRS, LIRIS, UMR5205, F-69621, Villeurbanne, France

²Univ Lyon, INSA Lyon, Université Claude Bernard Lyon 1, Université Lumière Lyon 2, DISP, EA4570, 69621 Villeurbanne, France

³Univ Lyon, INSA-Lyon, Université Claude Bernard Lyon 1, UJM-Saint Etienne, CNRS, Inserm, CREATIS UMR 5220, U1206, F-69100, LYON, France

⁴Univ Lyon, INSA Lyon, Inria, CITI, EA3720, 69621 Villeurbanne, France

⁵Univ Lyon, INSA-Lyon, CNRS, INRIA, LIRIS, UMR5205, F-69621, Villeurbanne, France

RÉSUMÉ

Face au constat d'une hétérogénéité grandissante des savoir-faire et connaissances en informatique des étudiants à l'arrivée en première année, et le risque de son exacerbation dans le contexte du « nouveau bac », nous avons voulu expérimenter une approche pédagogique, qui permette une gestion de cette hétérogénéité tout en respectant les contraintes d'un emploi du temps homogène et un coût constant. Les actions menées s'articulent autour de 4 pôles : la constitution de groupes de niveau, avec une attention particulière portée sur les 2 niveaux extrêmes (renforcement et avancé/en autonomie), la mise en place de QCMs réguliers, l'utilisation ponctuelle de l'Apprentissage Par Problème (APP), et un auto-positionnement. L'expérimentation est encore en cours, mais déjà de premiers éléments permettent d'ouvrir les échanges.

Mots-clés : hétérogénéité ; plate-forme d'apprentissage ; groupes de niveaux ; QCMs ; apprentissage par problèmes ; grille d'auto-positionnement ; autonomie ; pérennité des acquis

1 INTRODUCTION

L'observation de disparités croissantes en termes de savoir-faire et de connaissances en informatique des étudiants qui arrivent en première année à l'INSA Lyon, et la perspective de voir ces écarts se creuser davantage avec les "nouveaux bacheliers", sont à l'origine du projet présenté ici. L'objectif de ce projet était d'expérimenter différentes approches pédagogiques et outils tout en respectant les contraintes d'un emploi du temps homogène et un coût constant. Les options retenues devaient, de façon générale, permettre aux étudiants de progresser davantage à leur rythme, donner plus d'autonomie aux étudiants "avancés" et, côté enseignants, dégager du temps à consacrer aux étudiants les plus en difficulté.

La plupart des actions ont été menées dans une lanière de la filière classique et dans les filières internationales EURINSA et AMERINSA. Seuls le questionnaire d'auto-positionnement et le QCM final ont été étendus à toute la promotion de 1^{ère} année, pour un retour sur expérience. Dans la section 2, nous détaillons la mise en oeuvre de ces différentes actions. Un bilan sur chacune d'elles est présenté en section 3. Nous concluons sur un bilan plutôt positif et la proposition de quelques pistes d'amélioration.

2 LES DIFFÉRENTS ANGLES D'ATTAQUE ADOPTÉS DANS LE PROJET

Dans cette section, nous décrivons les différentes actions mises en place pour apprivoiser l'hétérogénéité des savoir-faire et connaissances en informatique des étudiants post-bac. Il est à souligner que tous les TDS en filière classique ont été déployés sur la plate-forme CaseInE¹, de même que ceux du premier semestre dans les filières EURINSA et AMERINSA. Cependant, ce choix n'a pas eu d'impact direct sur notre projet.

2.1 Groupes de niveau

Après un mois de cours et TDs communs en algorithmique et programmation, nous avons constitué des groupes de niveau afin de mieux gérer l'hétérogénéité des étudiants. Les groupes de niveau ont été mis en place à partir d'un QCM de positionnement composé de 12 questions séparées en 3 groupes. Les premières questions ont permis de recueillir l'auto-positionnement des élèves concernant leurs connaissances algorithmiques, leur aptitude à programmer ainsi que les langages de programmation qu'ils connaissaient déjà. Les questions suivantes permettaient d'évaluer 3 compétences distinctes : la compréhension de concepts clés en algorithmique (variable, for, if, while), la capacité de produire du code Java ou Python à partir d'un algorithme en pseudo code, et la capacité de proposer une solution algorithmique en pseudo-code, Java ou Python à partir d'un cahier des charges. À partir des résultats de ce QCM, nous avons réparti les étudiants en trois groupes de niveaux : un groupe en renforcement, un groupe classique et un groupe en autonomie.

Le fonctionnement adopté pour les groupes de niveau a été le suivant. L'encadrement et les supports de TDs ont été adaptés au niveau de chaque groupe en conservant un coût d'encadrement global constant. Pour le groupe "classique" aucun changement tant en encadrement qu'en forme du sujet. Pour le groupe "en autonomie", des questions de réflexions plus avancées ont été ajoutées aux sujets de TDs, et de nouveaux formats d'exercices ont été introduits afin de favoriser l'autonomie et le travail de groupe, ceci afin d'augmenter la motivation et l'implication des étudiants. Ainsi l'encadrant de ce groupe pouvait s'absenter sur la deuxième période du TD (2ème heure) pour participer à l'encadrement du groupe en renforcement. Pour le groupe en renforcement, des exercices très guidés ont été proposés afin d'introduire les notions de cours plus progressivement. Comme indiqué plus haut, l'encadrement des TDs a pu être dédoublé ponctuellement afin de mieux suivre les étudiants de ce groupe.

2.2 Travail régulier

L'informatique étant une discipline à faible coefficient du FIMI, il est parfois difficile de motiver les étudiants à travailler de manière régulière, chose pourtant indispensable à la réussite des étudiants les plus en difficulté. En particulier, les étudiants nous reprochent parfois, lors de l'évaluation des enseignements, d'utiliser des mots qu'ils ne comprennent pas. Ils ont aussi tendance à ne pas travailler suffisamment le cours avant d'arriver en TD, ou n'ont pas suffisamment conscience de la fragilité de leurs acquis pour poser des questions et ressentir le besoin de travailler. C'est pourquoi cette année nous avons introduit des QCM notés à chaque fin d'amphi. Nous avons utilisé Moodle comme support pour ces QCM, en privilégiant les questions de type

1. <https://moodle.caseine.org/>

"Choix multiples", "Appariement", "Glisser-déposer sur texte" et "Numérique". Ces QCM étaient courts (3 à 5 questions), et constitués de questions de cours construites pour une moyenne espérée de 17 ou 18 / 20.

2.3 Auto-positionnement

En milieu de 2^{ème} semestre, avant d'introduire la notion de programmation orientée objet, nous avons proposé aux étudiants de s'auto-positionner sur leurs compétences en algorithmique et programmation. Pour cela, nous avons constitué une grille comportant entre 3 et 4 niveaux pour chacune des 20 compétences cibles du programme.

Après s'être auto-positionnés sur cette grille, les étudiants ont passé un test surprise permettant d'évaluer leur niveau dans les 9 premières compétences (QCM de 30 questions). Chaque question était associée à une compétence et à un niveau. De plus, les étudiants ont pu se référer a posteriori à la grille d'auto-positionnement afin de confronter leur auto-évaluation à leurs compétences réelles.

Afin d'évaluer l'impact du projet sur les acquis, le test surprise a aussi été passé par les étudiants des autres groupes, pour identifier de potentielles différences de niveau.

2.4 Apprentissage Par Problème

L'objectif ici était d'expérimenter une démarche d'apprentissage inductif, qui amène les étudiants à travailler en groupe, avec un volant d'autonomie important visant à favoriser la réflexion (décomposition d'un problème, proposition d'un algorithme adapté, etc . . .), souvent bâclée par un recours trop rapide à la programmation. Pour cet APP nous avons choisi un ciblage très précis de l'activité. Elle s'est placée dans la progression pédagogique après une séquence portant sur les tableaux à une dimension, et a porté sur la découverte des tableaux à deux dimensions. Les compétences visées étaient la conception, le codage et les tests d'algorithmes de génération et de parcours de tableaux 2D; ainsi que, de façon plus générale, la compréhension et la réutilisation de code. Sur le plan des savoirs, le contenu a été délimité par une liste d'aspects techniques couverts, tels que par exemple : passage de paramètres, équation paramétrique, synthèse additive de couleurs. Ces objectifs en termes de compétences et de savoirs étaient communiqués aux participants.

Le problème posé aux étudiants dans cet APP était de réaliser la conception graphique d'un papier-peint pour la rénovation de leur chambre, et cela sous deux contraintes : 1) le dessin devait contenir un certain degré de répétition dans les motifs mais inclure également des variations ; 2) le dessin devait être généré par un programme à réaliser.

Pour le rendu final, il était demandé de soumettre un ou plusieurs des graphiques obtenus aux votes des autres étudiants, dans le but d'élire les plus appréciés. Le système de vote des pairs avait pour objet de fournir une motivation extrinsèque, et pour que cette motivation puisse être effective quel que soit le niveau des étudiants, nous avons choisi de faire élire les dessins dans des catégories humoristiques ne présumant pas du niveau informatique sous-jacent. En prévoyant plusieurs catégories (telles que la plus kitsch, la plus flashy, . . .), ceci permettait à tout un chacun de pouvoir espérer être nommé.

Concernant la motivation intrinsèque, cet APP nous permet de nous appuyer sur l'ignition fournie par la sollicitation de la créativité des participants sur le plan de la conception graphique. Elle apporte également la possibilité de préserver la motivation des étudiants les plus aguerris, en leur offrant une latitude importante d'investigations techniques. Enfin, pour maintenir la dynamique de réflexion personnelle au fil de l'APP, nous avons favorisé le partage d'idées en faisant travailler les étudiants en binôme.

Les supports préparés comportaient plusieurs volets. Lors de la première séance, une fiche descriptive de mise en situation a été remise aux étudiants, et l'enseignant a fait une courte intervention pour initier la démarche. Les supports comportaient aussi des lots d'indices à communiquer aux étudiants selon les besoins et l'avancement. Ces lots étaient constitués de schémas, de liens vers des sources d'information sur internet, et d'exemples de programmes en rapport avec le problème à résoudre. Ces exemples avaient été conçus pour permettre aux étudiants de découvrir progressivement des aspects techniques pouvant être utiles dans la résolution du problème.

Cet APP s'est déroulé sur trois semaines à raison de deux heures de travail encadré sur machine par semaine et environ autant de travail personnel attendu en dehors des séances. En plus du rendu final des dessins soumis aux votes, chaque binôme devait envoyer à son enseignant des échantillons produits à la fin des deux premières séances. Ces remises intermédiaires ont été importantes pour cadencer l'avancée des étudiants dans leur travail, et bien sûr pour permettre aux enseignants d'avoir une vision globale de la progression.

3 BILANS DE L'EXPÉRIENCE

Dans cette section, nous présentons un bilan des différentes actions décrites précédemment.

3.1 Bilan sur les groupes de niveau

Des groupes de niveau avaient été mis en place depuis 3 ans, mais avec un objectif très différent. En effet, ils étaient initialement un moyen d'étude et de lutte contre les *a priori* sur l'informatique liés au phénomène de stéréotype de genre. À l'inverse, cette année, nous voulions générer 3 niveaux, avec pour objectif de libérer du temps enseignant dans le groupe avancé pour le ré-investir dans le groupe en difficulté. Contrairement aux années passées, le test utilisé pour créer les groupes s'est déroulé après le début des enseignements de Java, et a donc pu porter sur des questions de Java (et non pas sur des *a priori*). Lors du test, nous avons demandé aux étudiants d'estimer leur niveau, tandis qu'après le semestre nous leur avons demandé dans quel groupe ils auraient du être selon eux, et dans quel groupe ils pensaient qu'ils se seraient placés si on le leur avait demandé au lieu de faire le test. Les résultats (Table 1) montrent que le test a aidé, mais que cela reste perfectible. En revanche, si l'on compare le groupe "idéal" dans lequel ils se placent à leur déclaration initiale de niveau de compétence, alors les résultats semblent totalement aléatoires. Il est donc important de ne pas se fier à ce point, pas plus qu'au nombre de langages qu'ils auraient vu avant de venir à l'INSA, ou encore à leur score Pix de la compétence « Programmer ».

Bien que perturbés par le confinement, les groupes de niveau tels que mis en place sont plébiscités par les étudiants. Ils ont notamment apprécié que les sujets soient plus ou moins guidés, et avec des discussions plus ou moins poussées, leur permettant de réduire le décrochage ou l'ennui en séance. Le principal frein à cette démarche, en revanche, est sa lourdeur de mise en oeuvre, en l'absence de soutien administratif (listings, absences, EdT, ...), ce qui empêche d'avoir de la souplesse. Ainsi, quelques étudiants mal placés ont été contraints de rester dans leur groupe de niveau par manque d'agilité du processus. Nous n'avons pas repéré d'étudiant ayant chuté sous la moyenne en ayant été mis dans le groupe avancé, et seul 2 étudiants placés dans le groupe intermédiaire ont eu entre 9 et 10. Nous avons d'autre part brièvement échangé avec 4 étudiants qui avaient été mis dans le groupe "renforcé", et qui ont très bien réussi leurs évaluations au semestre 1, ainsi que les QCMs du semestre en cours. Bien que tous ne se soient pas retrouvés dans ce groupe pour les mêmes raisons (3 ont dit ne pas avoir eu de base en algorithmique, alors que le quatrième a parlé de difficultés à travailler sur un ordinateur), globalement, tous ont affirmé que le fait de se retrouver dans le groupe de renforcement leur a été bénéfique, entre autre parce qu'ils ont ainsi été mis en confiance et ont pu progresser à leur rythme. Quand ils ont été assez "aguerris", ils ont eu moins recours aux enseignants, et ont de surcroît fait les questions "bonus" des sujets de TD (du groupe renforcé). Donc, en fin de compte, aucune de nos erreurs potentielles de placement n'a eu de conséquence dramatique.

Enfin, on peut constater un écart plutôt positif sur les résultats de l'évaluation de fin de premier semestre entre la lanière expérimentale en classique et les autres lanières classiques et ASINSA (Table 2), même s'il n'est pas significatif.

	Nous avons ...	Ils se seraient ...
Sous-évalué	18%	27%
Correctement placé	71%	66%
Sur-évalué	11%	7%

TABLE 1: Analyse positionnement (déclaratif)

	Lanière expérimentale (85 étudiants)	Autres lanières classiques & ASINSA (416 étudiants)
Moyenne	12.22	11.78
Médiane	12.00	11.63
Ecart-type	4.10	4.26
nombre inf à 7	9.4%	12.5%
nombre inf à 10	31.8%	35.1%

TABLE 2: Analyse quantitative de l'IE de fin de premier semestre en classique et ASINSA

3.2 Travail régulier

Selon les groupes, les QCMs se sont tenus soit en fin d'amphi, soit en début du premier TD qui suivait l'amphi. Les étudiants des filières EURINSA et AMERINSA, qui ont vécu les 2 situations, ont largement plébiscité l'option début de TD, qui leur permettait de "travailler leur cours avant le QCM". Pour les étudiants en filière classique, la question n'a pas pu être posée, car les TD étaient à la suite directe du cours.

Le confinement ne nous ayant pas facilité la tâche, les QCM ont été faits en distanciel, et quand il y a eu des problèmes techniques, cela n'était pas très simple à gérer. Cependant, les résultats ont été très encourageants. Les moyennes sont hautes (bien que plus basses que celles attendues, par exemple 15.78 pour les classiques), ce qui motive les étudiants. De plus, les moments de correction ont permis d'échanger sur les difficultés, transformant l'évaluation en un moment d'apprentissage.

Le bilan s'avère donc très positif pour cette modalité, à quelques détails près. Chaque enseignant faisant un amphi un peu différent, et les créneaux étant sur des horaires décalés, il a été nécessaire pour chaque enseignant d'amphi de créer lui-même l'ensemble des questions relatives à son amphi, dans un temps assez court. Cette absence de répartition de charge a eu un impact non négligeable, et les questions contenaient assez souvent des typos ou des erreurs. Nous avons donc régulièrement dû "offrir" les points de certaines questions aux étudiants.

3.3 Auto-positionnement et QCM

Le QCM post auto-positionnement a aussi bien été déployé auprès des étudiants de la manière expérimentale classique que des autres manières classiques. Une première analyse quantitative rapide ne permet pas de voir une différence significative entre les deux populations (Table 3), malgré une moyenne et une médiane légèrement supérieure pour la manière expérimentale.

De plus, l'analyse détaillée pour chaque étudiant des écarts entre l'auto-positionnement et le résultat du QCM n'a pas encore été faite. Cependant, sur les 5 cas qui ont pu être observés, les deux grilles sont assez similaires, avec une légère sous-évaluation par l'étudiant pour certaines notions. Enfin, les retours des étudiants à l'oral ont fait état d'un intérêt pour ce QCM car il a à la fois permis de faire un point de révision sur les différentes compétences visées jusque là, et il a aussi rassuré sur le niveau attendu.

Le QCM de positionnement reste perfectible. Ainsi, une seule question par niveau et par notion a été posée, avec un risque de mauvaise compréhension et/ou rédaction de la question. Ainsi, on peut constater par exemple que sur la notion Tableau 1D, les étudiants ont eu de meilleurs résultats en moyenne au niveau 3 ("Je sais comment est stocké en mémoire un tableau", moyenne 0.79 / 1) qu'au niveau 1 ("Je sais déclarer et initialiser un tableau", moyenne 0.62 / 1).

	Lanière expérimentale (68 répondants)	Autres lanières classiques (319 répondants)
Score moyen (/30)	21.55	21.15
Score médian (/30)	21.95	21.27
Ecart-type	3.78	4.09

TABLE 3: Analyse quantitative du QCM de positionnement

3.4 Apprentissage par problème

Le retour des étudiants à l'oral, après l'APP, a fait état de l'intérêt qu'ils ont eu pour la démarche et de leur adhésion. Un seul bémol est à mentionner de leur point de vue : il s'agit d'une légère inquiétude de certains quant au fait qu'ils n'ont pas suivi les trois séances classiques faites par les autres groupes à ce moment-là (alors que les évaluations à venir seront communes à tous les groupes).

Concernant les enseignants, bien que l'APP leur ait demandé un investissement conséquent, la transition sur le plan pédagogique a été facilitée par la présence dans l'équipe de plusieurs membres ayant déjà pratiqué ce type d'approche par problèmes.

4 CONCLUSION ET PERSPECTIVES

De façon générale, les quatre actions mises en oeuvre dans ce projet semblent avoir eu un impact plutôt positif sur la motivation, l'apprentissage et la progression des étudiants. En particulier, les approches duales TDs différenciés, et session d'APP proposant un sujet de travail sans borne maximale, comme outils de gestion de l'hétérogénéité des pré-acquis et des rythmes d'apprentissage, ont été toutes deux plébiscitées par les étudiants. Il reste bien entendu des points à améliorer dans les dispositifs, comme par exemple : affiner la procédure de constitution des groupes de niveau; développer plus avant l'utilisation de Caseine pour proposer aux étudiants avancés des exercices autoévalués (jeux de tests) pour renforcer la prise d'autonomie; amender la grille d'auto-positionnement, et étendre son utilisation à l'ensemble de l'année pour aider les étudiants à identifier leurs potentielles difficultés. Enfin se pose aussi, à moyen terme, la question du passage à l'échelle de tout ou partie des actions déployées.