

Diagnostics combinés pour l'adaptation dans un environnement de résolution de problèmes complexes

Cédric d'Ham¹, Isabelle Girault¹, Claire Wajeman¹, Agnès Berthet¹, Laure Piron¹

¹ Université Grenoble Alpes, CNRS, Grenoble INP, LIG, F-38000 Grenoble, France
Contact : cédric.dham@univ-grenoble-alpes.fr

1 Contexte : le logiciel TitrAB

En 2011, les programmes de lycée ont introduit de nouvelles compétences exigibles des élèves dans le domaine des sciences expérimentales. L'épreuve d'« Evaluation des Capacités Expérimentales » du baccalauréat détermine ces attendus. Les élèves doivent notamment être capables d'élaborer le protocole d'une expérience scientifique. Pour l'acquisition de cette capacité, nous avons développé le logiciel TitrAB (<http://titrab.imag.fr/>) [1] qui propose 16 exercices d'entraînement à la conception d'une expérience de titrage acido-basique. Dans chaque exercice, l'apprenant choisit les valeurs des paramètres de l'expérience (cf. fig. 1) qui est ensuite exécutée par un simulateur intégré. Dans le cas où les valeurs choisies ne permettent pas de simuler l'expérience, un système de tuteur intelligent fournit un message de rétroaction adapté.

The screenshot displays the 'Dispositif expérimental' (Experimental Device) configuration screen. It features a central diagram of a titration setup with a burette, a beaker, and a pH meter. The interface includes several input fields and dropdown menus for defining the experiment's parameters. At the top, it asks to fill a 25 mL burette with a specific solution, with options for 'non dilué(e)' or 'dilué(e) auparavant'. Below this, a 'Dilution' section allows for specifying the volume of solution to be introduced, the volume of solvent, and the final volume. A similar section is provided for the 100 mL beaker. At the bottom, there are fields for adding additional reagents and a button labeled 'Faire les ajouts et relever les mesures'.

Fig. 1 : espace d'élaboration du protocole de titrage dans le logiciel TitrAB

Les activités proposées aux apprenants dans TitrAB sont des problèmes de conception qui font partie des problèmes complexes. Cette complexité tient à plusieurs caractéristiques :

- bien que l'interface proposée pour l'élaboration de l'expérimentation soit fermée (cf. fig. 1), la combinatoire des solutions est élevée : suivant les exercices, il peut y avoir 39 jeux de valeurs simulables et une infinité de jeux de valeurs non simulables ;
- pour un exercice, il n'existe pas forcément de solution unique : plusieurs jeux de valeurs peuvent convenir ;
- concevoir une expérience nécessite d'anticiper ses résultats, ce qui demande de maîtriser finement les concepts théoriques associés.

16 exercices de niveaux croissants sont proposés dans TitrAB. Le niveau des exercices est déterminé à partir de quatre variables didactiques [2] comme, par exemple, le « rapport de concentration des solutions titrantes et titrées ». TitrAB est utilisé depuis 2016 avec environ 500 apprenants par an, au niveau lycée et licence.

2 Diagnostics et adaptation dans TitrAB

L'adaptation dans TitrAB est envisagée à travers deux mécanismes : l'affichage d'un message adapté en cas d'expérience mal construite et la proposition de parcours d'exercices personnalisés parmi les 16 exercices disponibles dans l'environnement. Le premier mécanisme (messages) est en partie implémenté et le second (parcours) est en phase de conception.

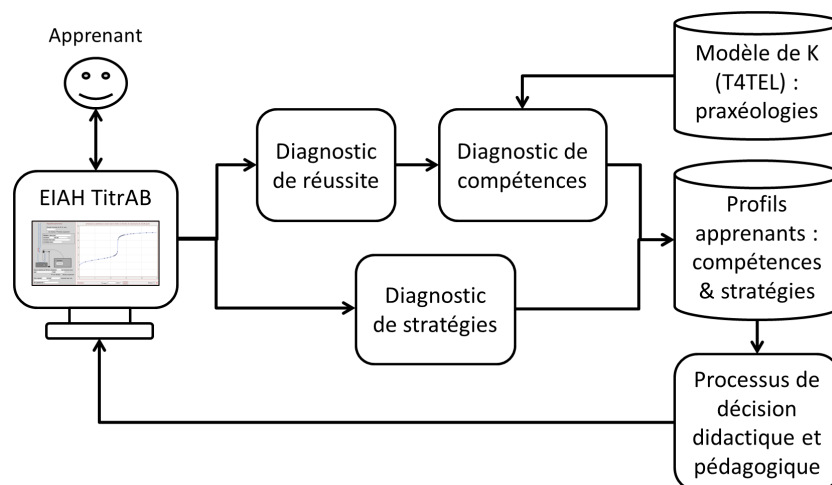


Fig. 2 : processus d'adaptation dans TitrAB

Le processus de décision didactique et pédagogique (PDDP) a pour fonction de calculer la rétroaction du système : message ou proposition d'un nouvel exercice. Cette décision s'appuie sur des résultats issus de diagnostics. Dans TitrAB, trois modes de diagnostics sont envisagés (cf. fig. 2). La description de ces diagnostics (réalisés ou en cours de réalisation) est l'objet de cette contribution.

Le premier diagnostic évalue la réussite de l'étudiant et se situe au niveau comportemental [3]. Il est effectué chaque fois que l'étudiant soumet un protocole au système afin qu'il soit simulé. Ce diagnostic consiste à appliquer un ensemble de doubles contraintes

[pertinence ; satisfaction] sur l'état de la production soumise [4]. Les contraintes sont définies par les concepteurs à partir d'une analyse experte des réponses valides. Le résultat du diagnostic de réussite est une liste de *succès & erreurs* caractérisant la production d'un étudiant à un instant précis et qui est enregistrée comme trace du système.

Le diagnostic de compétences utilise les succès & erreurs de l'étudiant déterminés précédemment. Ce diagnostic se situe au niveau épistémique dans le sens où il n'évalue pas la production de l'apprenant mais ses capacités à résoudre le problème posé [3]. Le diagnostic de compétences s'appuie sur un modèle de connaissance que nous représentons sous forme d'une praxéologie et décrivons avec le formalisme du modèle T4TEL [5]. En résumé, la connaissance utile pour résoudre le problème de conception d'expérience est décrite par des quadruplets praxéologiques [type de tâche ; technique ; technologie ; théorie] [6]. Pour un type de tâche donné T_1 , une technique valide est un ensemble de types de tâches permettant de résoudre T_1 ; la technologie permet de justifier et adapter la technique ; la théorie justifie la technologie hors du contexte de T_1 . Pour notre diagnostic de compétences, chaque succès ou erreur, issu du diagnostic de réussite, a été relié à un ou plusieurs types de tâche(s) de conception. Un système de calcul probabiliste sera mis en place afin de déterminer, à partir de l'historique de ses succès et erreurs, dans quelle mesure l'apprenant maîtrise les techniques permettant de résoudre avec succès les types de tâches de conception. Ce diagnostic est incrémental et son résultat sera stocké dans une base de profils d'apprenants, sous la forme de *compétences* acquises, avec un degré de certitude estimé.

Le troisième diagnostic utilise une voie différente des deux premiers. Il est basé, non plus sur l'état des productions des apprenants, mais sur leurs traces d'activités enregistrées dans TitrAB. L'objectif de ce diagnostic est de détecter les *stratégies* mises en place (consciemment ou non) par l'apprenant pour résoudre le problème de conception dans TitrAB. Les stratégies que nous cherchons sont, soit décrites dans la littérature, *e.g.* l'essai-erreur systématique (« gaming the system ») [7] ou la recherche abusive d'aide [8], soit définies à partir de notre expérience, *e.g.* l'abandon de la tâche ou l'exploration aléatoire du dispositif. Les stratégies des étudiants ont un caractère temporel localisé : le diagnostic cherche donc à détecter des *motifs* (« patterns ») d'activité sur une fenêtre de temps glissante assez courte. Ces motifs de traces sont des comportements de l'apprenant que nous définissons comme étant caractéristiques des stratégies employées. Pour définir les motifs à rechercher, nous utilisons des techniques de visualisation des traces qui font apparaître les caractéristiques que nous estimons saillantes dans la définition d'une stratégie. Par exemple, pour définir un motif représentatif de la stratégie « essai-erreur systématique », nous cherchons les traces dans lesquelles l'apprenant modifie le même paramètre plusieurs fois de suite dans des temps assez courts, en demandant une validation après chaque modification. La visualisation des traces permet de valider l'existence de tels motifs dans les traces. Le lien effectif avec la stratégie recherchée sera à valider en conditions écologiques une fois le diagnostic mis en place. Les stratégies détectées seront enregistrées dans le profil de l'apprenant.

Actuellement, seul le diagnostic de réussite est implémenté dans TitrAB et alimente le PDDP. Celui-ci, très simple, affiche des messages qui décrivent les erreurs détectées dans le protocole de l'apprenant. A terme, les résultats des trois diagnostics ainsi que des données comme l'avancement du travail de l'étudiant seront combinés pour permettre de calculer une rétroaction tenant compte de ces différents éléments [9]. Nous donnons ici quelques principes directeurs pour ce futur développement. La détection d'une erreur dans le protocole (contrainte non vérifiée) déclenchera toujours l'affichage d'un message pour l'apprenant. Ce message dépendra du type de tâche auquel est rattachée l'erreur, mais aussi des messages obtenus précédemment pour ce type de tâche, de la compétence de

l'apprenant estimée pour ce type de tâche, de la stratégie employée par l'apprenant au moment de la rétroaction et de son avancement dans la résolution du problème. Les différents messages pouvant être envoyés seront choisis dans une banque de messages formulés à différents niveaux de description de l'erreur : erreur, type de tâche, technique, technologie ou théorie. D'autres messages, formulés à un niveau méta-cognitif, pourront aussi être affichés, si la stratégie employée par l'étudiant est diagnostiquée comme étant néfaste pour ses apprentissages. Par ailleurs, un nouveau type de rétroaction sera implémentée de façon à adapter les parcours d'études des apprenants. Le système proposera aux apprenants de travailler en premier lieu, sur l'exercice le plus simple, puis sur des exercices de difficulté adaptée à leur performance et à leur maîtrise des compétences en jeu.

La complexité du PDDP va ainsi augmenter de façon importante mais devrait permettre de fournir des rétroactions adaptées, non seulement à l'état du protocole soumis par l'apprenant (comme actuellement), mais aussi à l'activité passée de l'apprenant : les rétroactions reçues, ses compétences et les stratégies employées.

Références

1. Berthet, A., d'Ham, C., Girault, I.: TitrAB : un logiciel pour apprendre à élaborer le protocole d'un titrage acido-basique. *Bulletin de l'Union des Physiciens*. 111, 999–1012 (2017).
2. d'Ham, C., Girault, I., Berthet, A.: Faire apprendre les sciences par l'élaboration de protocoles expérimentaux : modèles et étayages produits lors de la conception d'une séquence pratique sur les titrages acide-base. *RDST*. 19, A paraître (2019).
3. Balacheff, N.: Exigences épistémologiques des recherches en EIAO. *Revue d'Ingénierie Educative*. 4–14 (1992).
4. Ohlsson, S.: Constraint-Based Student Modeling. In: Greer, J.E. and McCalla, G.I. (eds.) *Student Modelling: The Key to Individualized Knowledge-Based Instruction*. pp. 167–189. Springer Berlin Heidelberg (1994).
5. Chaachoua, H.: T4TEL un cadre de référence didactique pour la conception des EIAH. In: *Pré-Actes du séminaire de didactique des mathématiques - 2-3 février 2018*. pp. 5–22. Julia Pilet & Céline Vendeira, Paris (2018).
6. Girault, I., Wajeman, C., d'Ham, C.: Modèle de construction d'un EIAH pour une activité de conception expérimentale. In: *6e Congrès international sur la théorie anthropologique du didactique*. , Autrans, France (2018).
7. Baker, R.S.J. d., Corbett, A.T., Roll, I., Koedinger, K.R.: Developing a generalizable detector of when students game the system. *User Model User-Adap Inter.* 18, 287–314 (2008). <https://doi.org/10.1007/s11257-007-9045-6>.
8. Roll, I., Baker, R.S.J. d., Aleven, V., Koedinger, K.R.: On the Benefits of Seeking (and Avoiding) Help in Online Problem-Solving Environments. *Journal of the Learning Sciences*. 23, 537–560 (2014). <https://doi.org/10.1080/10508406.2014.883977>.
9. Murray, R.C., VanLehn, K., Mostow, J.: Looking ahead to select tutorial actions: A decision-theoretic approach. *International Journal of Artificial Intelligence in Education*. 14, 235–278 (2004).