



HAL
open science

**ETM ET ACTION DIDACTIQUE CONJOINTE:
regards croisés sur l'enseignant intégrant une tâche de
programmation informatique en classe de
mathématiques**

Mathématiques Fabienne Venant, Michèle Couderette

► **To cite this version:**

Mathématiques Fabienne Venant, Michèle Couderette. ETM ET ACTION DIDACTIQUE CONJOINTE: regards croisés sur l'enseignant intégrant une tâche de programmation informatique en classe de mathématiques. Huitième Symposium d'Étude sur le Travail Mathématique (ETM8), Oct 2024, Castro Urdiales, Espagne. hal-04817673

HAL Id: hal-04817673

<https://hal.science/hal-04817673v1>

Submitted on 3 Dec 2024

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.

ETM ET ACTION DIDACTIQUE CONJOINTE: REGARDS CROISÉS SUR L'ENSEIGNANT INTEGRANT UNE TÂCHE PROGRAMMATION INFORMATIQUE EN CLASSE DE MATHÉMATIQUES.

Fabienne Venant et Michèle Couderette

Université du Québec à Montréal et IRL CRM-CNRS, Université de Montréal,
LDAR, Université Paris-Est Créteil

Nous nous intéressons à la façon dont un enseignant non expert conçoit et met en œuvre une tâche de programmation informatique dans le cadre de son enseignement des mathématiques. Nous analysons un épisode dans une séquence d'enseignement intégrant la programmation informatique dans une classe de mathématiques secondaire au Québec. Nous convoquons à la fois la théorie des ETM et le modèle de l'action didactique conjointe (ADC). Nous discuterons des contributions mutuelles de ces cadres théoriques, ainsi que des enjeux didactiques de l'intégration de la programmation informatique à l'école.

CONTEXTE

Le Québec, dans la lignée d'autres pays à l'échelle mondiale, discute depuis quelques années de la pertinence de réintroduire la programmation informatique dans les curricula scolaires. Le rapport Barma (2018) recommande que la programmation informatique soit ajoutée au curriculum québécois comme une nouvelle compétence transversale en lien avec la pensée informatique. Il convient donc de s'interroger sur la nature de l'articulation entre travail mathématique, travail algorithmique et travail informatique (Venant et al., 2023). Des recherches ont montré la nécessité d'une formation des enseignants visant à la fois des savoirs savants et des savoirs pour enseigner (Briant, 2013; Couderette, 2016, 2023). Des questions se posent également sur l'accompagnement des enseignants dans la conception d'activités intégrant la programmation informatique (Venant, 2023). C'est dans ce contexte que nous avons mené un projet de recherche¹ visant à développer et expérimenter de nouveaux modes de formation auprès des enseignants cherchant à intégrer la programmation informatique en classe de mathématiques. Nous proposons ici l'analyse de la mise en œuvre, par l'une des participantes au projet, d'une tâche intégrant la programmation informatique. Cette tâche a été conçue de façon collaborative dans le cadre du projet, avec l'intention didactique d'articuler travail informatique et travail mathématique.

¹ Projet « Intégration de la programmation en mathématiques et en sciences au secondaire: quels enjeux pour la formation des enseignants » financé par le ministère de l'Éducation du Québec (MEQ) et le Fonds de recherche Québec" (FRQ)

CADRES THÉORIQUES

Notre analyse s'appuie sur une articulation du modèle de l'action didactique conjointe (Schubauer-Leoni et al., 2007; Sensevy, 2007) et de la théorie des Espaces de Travail Mathématiques (ETM, Kuzniak, 2011).

Le modèle de l'action didactique conjointe

Le modèle de l'action didactique conjointe (Schubauer-Leoni et al., 2007; Sensevy, 2007) permet de rendre compte de la co-construction du savoir au sein d'un système didactique : les apprentissages en classe sont le produit de l'action didactique conjointe professeur élèves (Ibid.) et émergent lors des interactions en classe. Aussi, nous utilisons le triplet des genèses {meso/topo/chronogenèse} pour rendre compte de la dynamique évolutive du système didactique. Dans le cadre de cette communication nous ne pointerons principalement que les positions relatives de l'enseignante et des élèves (tonogenèse²) et de leur incidence sur l'avancée du savoir (chronogenèse) tout au long de l'action didactique. L'agir professoral est, quant à lui, analysé au travers des descripteurs « définir », « dévoluer », « réguler » et « institutionnaliser ». *In fine*, il s'agit de déterminer le savoir réellement construit par les élèves.

Ces analyses sont articulées avec celles du travail mathématiques et algorithmique, au sein de la théorie des ETM.

La théorie des espaces mathématiques de travail

Dans la lignée de Lagrange et Laval (2023), nous considérons un ETA (espace de travail algorithmique). Nous incluons dans le référentiel théorique les instructions, les structures de contrôles et les processus algorithmiques comme l'affectation ou l'itération. Les *representanem* réfèrent à des instructions et des opérateurs algorithmiques pouvant également être manipulés à travers des artefacts logiciels (plateforme informatique) ou tangibles (papier-crayon, jeu de LEGO...). La genèse discursive de l'ETA préside la construction d'un algorithme à partir des éléments du référentiel théorique et met en jeu des raisonnements itératifs ou récursifs. L'ETA permet d'appréhender l'articulation entre travail mathématique et travail algorithmique (Venant et al., 2022). Nous intéressons ici aux Espaces de Travail Idoines effectifs, qui permettent de repérer ce qu'un enseignant décide de privilégier ou d'exclure autour d'un problème relativement à la perspective de mener un travail en classe (Masselin, 2019). L'idonéité cherche à capturer le processus d'adaptation et de raffinement, donc de l'intelligence, en classe. Elle s'inscrit aussi bien dans

² La topogenèse éclaire l'action des transactants (élèves et enseignant) engagés dans l'action didactique. Elle décrit les positions relatives des élèves et du professeur dans la gestion de l'activité. Élèves et enseignant interagissent sur les objets de savoir, les faisant évoluer tout au long du temps didactique. La topogenèse rend ainsi compte de la manière dont les acteurs partagent les responsabilités dans la manipulation des objets de savoirs, l'un dans son rôle d'enseignant, l'autre dans son rôle d'élève.

l'interaction sujet-milieu que dans l'interaction entre l'enseignant et le système sujet-milieu de Brousseau (Bruillard et Richard, 2024)

PRESENTATION DE LA TÂCHE

Labyrinthe algébrique.

L'enseignante, Noémie, présente aux élèves un « labyrinthe algébrique » (fig.1) sur la plateforme en ligne Scratch³. Un lutin *chat* doit se déplacer vers la sortie de ce labyrinthe en ramassant les objets disséminés. Pour acquérir chaque objet, il est nécessaire de résoudre une équation générée aléatoirement et rendue visible au contact du lutin avec l'objet.

Tant que l'équation n'est pas résolue, une autre équation est générée. La sortie du labyrinthe est validée par un bouton compteur qui vérifie que l'ensemble des objets a bien été ramassé.

Le labyrinthe présenté aux élèves ne contient que deux lutins objets (*cloche* et *cadeau*). Après analyse des programmes existants, l'élève doit compléter le jeu en insérant trois autres lutins (*clé*, *trompette* et *bonnet*) générant de nouvelles équations.

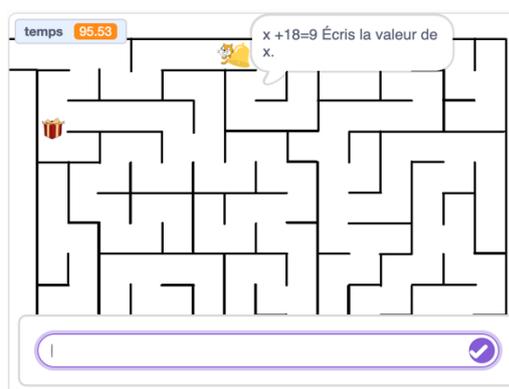


Figure 1: labyrinthe algébrique

A chaque objet est associé un niveau de résolution d'équation. Ainsi, le sous-programme « cadeau » génère⁴ une équation de niveau 1, de type $a + x = b$. L'équation de la forme $ax = b$ (niveau 2) est générée par le lutin *cloche*. Les équations de la forme $ax + b = c$ ou $x/a + b = c$ (niveau 3) sont générées par les lutins *trompette* et *bonnet*. Enfin, les équations de la forme $ax + b = cx + d$ (niveaux 4) sont générées par le lutin *clé*. Anticipant des difficultés, l'enseignante a découpé la séance en étapes qu'elle considère progressives et porteuses d'autonomie : « *Je vais vraiment par étapes. Au début, c'est analyser le programme [...] On a vraiment planifié pour y aller [en programmation]. Pour quelqu'un qui n'y connaît rien, il y a un programme, il faut juste qu'il change quelques blocs. Après ça, oups, c'est un petit peu plus d'autonomie, et puis c'est ça, on continue.* » La tâche est ainsi découpée en sept parties. Les trois premières étapes consistent à analyser le programme du lutin *chat* (ses déplacements), écrire le programme du lutin *bouton-compteur* (valider la sortie), écrire le programme du lutin *cadeau* (générer aléatoirement une équation de

³ Les enseignants peuvent déposer leurs projets Scratch sur une plateforme accessible à tous : <https://scratch.mit.edu/projects>

⁴ Par commodité de langage, nous écrirons que les différents lutins (cadeau, cloche, etc.) génèrent une équation.

niveau 1). Les quatre suivantes demandent l'écriture des programmes des *lutins* supplémentaires (*bonnet, clé, trompette*).

Travail potentiel

La tâche est très complexe. On peut la décomposer en trois sous tâches :

1. Génération d'équation : chacun des lutins *objet* doit générer une équation du premier degré répondant à certaines contraintes sur le coefficient, s'assurer que cette équation possède bien une solution entière, afficher cette équation et en demander la solution au joueur.
2. Résolution d'équation : chacun des lutins *objet* doit calculer la solution de l'équation précédemment générée, et comparer la valeur calculée et la valeur entrée par le joueur.
3. Gestion du labyrinthe : cette tâche est répartie entre les lutins *chats* et *bouton-compteur*. Elle inclut les déplacements du lutin *chat* grâce aux commandes *flèche*, la gestion de la taille des lutins, des sons qu'ils émettent et des temps d'attente, la prise en compte de la topologie du labyrinthe (ne pas traverser les murs...), la comptabilisation du nombre de trésors gagnés et la gestion du chronomètre.

Nous distinguons dans ces tâches trois type de travail. Le travail **instrumental** concerne les aspects techniques du design du jeu. Ce travail concerne, par exemple, la gestion des événements (du type *quand une flèche/une couleur est touchée*), des instructions conditionnelles et des tests d'égalité (Si *Trésor n'est pas égal 5*). Le travail **algorithmique** supporte la mise en place des structures itératives plus complexes. Il inclue le choix et l'affectation des variables, la définition des relations entre les variables, la détermination des opérateurs pertinents pour le respect des contraintes mathématiques, l'élaboration des tests d'entrée et de sortie des itérations. Le travail **mathématique** permet de déterminer de façon générique les solutions des équations générées Il permet également de traduire sous forme algébrique les contraintes numériques sur le coefficient et la solution de l'équation qui président à la génération des équations. Ce travail mathématique repose sur une capacité à généraliser la résolution d'équations du premier degré. Il met en jeu un raisonnement analytique (Radford, 2014) permettant de traiter les quantités indéterminées que sont les coefficients et l'inconnue de l'équation comme si elles étaient connues, et à parvenir à réaliser des opérations sur ces nombres inconnus. Cette articulation passe notamment par une phase de dénotation (Radford, 2014) permettant à d'associer variables informatiques et symbolisation des nombres inconnus.

ANALYSES

La séance est rythmée selon une même modalité pour chacune des étapes : une phase de dévolution en binôme puis une phase de régulation. D'un point de vue transpositif et au regard des enjeux épistémiques de la séance, nous observons ici une élémentarisation des savoirs attendus permettant par le franchissement d'étapes, de

maintenir le sens global de l'activité. Dans cette communication, nous nous centrons sur la troisième étape consistant à analyser le programme du cadeau : générer aléatoirement une équation de forme $x + a = b$, demander sa solution puis mettre à jour le compteur.

Le milieu didactique comprend les éléments constitutifs des milieux précédents (programmation des déplacements des lutins *chat* et *bouton-compteur*) augmenté du programme (Fig. 2) du lutin cadeau générant aléatoirement des équations de la forme $x + a = b$, puis demandant et vérifiant une solution au joueur. Ce programme engage des savoirs mathématiques non connus des élèves (la généralisation d'une équation et de sa solution). Il convoque également des concepts informatiques non triviaux : le concept de variable et de son initialisation, les structures de contrôles, la gestion d'une boucle *jusqu'à*. Il articule de façon complète et complexe le travail mathématique et le travail algorithmique. Généralisation et raisonnement itératif se nourrissent mutuellement

Bien que le milieu didactique à analyser soit particulièrement riche, l'enseignante laisse peu de temps aux élèves : « *Je te donne six minutes. Tu as beaucoup moins de temps, on se dépêche pour analyser la partie du cadeau* » précisant que « *C'est là que la partie mathématique embarque* ». Cette dernière précision laisse suggérer que le *prima* est donné aux mathématiques. Nous faisons l'hypothèse que l'équation étant de niveau 1 (de la forme $x + a = b$), l'enseignante considère le traitement de sa généralisation comme simple et accessible aux élèves et de ce fait, ne nécessitant pas un temps d'analyse important. En termes d'ET, on peut interpréter ce choix comme un accent mis sur le travail dans l'ETA, la fibration vers l'ETM via le travail de généralisation étant considérée comme quasi-instantanée.

```

i1 quand est cliqué
i2 attendre 1 secondes
i3 metre variable à 1000
i4 répéter indéfiniment
i5 si touche le joueur ? alors
i6 dire Voici un équation à résoudre. pendant 2 secondes
i7 répéter jusqu'à ce que variable = terme constant à droite - terme constant à gauche
i8 metre terme constant à droite à nombre aléatoire entre 2 et 20
i9 metre terme constant à gauche à nombre aléatoire entre 2 et 20
i10 demander regrouper regrouper x+ et regrouper terme constant à gauche et regrouper = et regrouper terme constant à droite et Écris la valeur de x. et attendre
i11 metre variable à réponse
i12 cacher
i13 jouer le son guitar strum
i14 ajouter 1 à trésor
i15 stop ce script
  
```

Figure 2 : Partie 3 - Analyser le programme du cadeau

1er épisode : Une confusion entre variable mathématique et variable informatique

Lors de la phase de dévolution de cette troisième étape, l'enseignante opte pour une position topogénétique relativement basse : elle circule de binôme en binôme, questionnant parfois, guettant l'avancée des élèves. Nous l'observons happée par des questionnements d'ordre algorithmique : les élèves incités par les réglures du document-élève à la suite de chaque instruction (cf. figure 2), se maintiennent dans le droit fil des étapes précédentes, étudiant le programme ligne à ligne ce qui, dans un premier temps, les maintient sur un versant informatique de l'activité. Le travail se situe donc au sein de l'ETA. L'analyse est purement séquentielle et confine le travail dans le plan sémiotique-instrumental. Noémie facilite d'abord une genèse instrumentale, à travers la manipulation, au sein de la plateforme, des artefacts constitués par les « blocs » scratch : « *Si tu ne sais pas à quoi sert un bloc, tu peux aller modifier le programme pour essayer de comprendre* ». Cette analyse bloc par bloc soutient une genèse sémiotique, médiée par les blocs représentant les objets algorithmiques. C'est en tant qu'étiquettes que les instructions et objets algorithmiques s'instancient dans le plan cognitif. La construction de leur sens est partielle et leur permet difficilement d'acquérir un statut discursif. L'échange ci-dessous montre, par exemple, une difficulté à appréhender la notion de variable et son initialisation.

Élève: La variable, c'est quoi? parce qu'on a de la misère à comprendre ...

Ens.: Dans une équation, mettons $x + 25 = 40$. Ma variable, ça va être mon x

Élève: ah ? ok..

L'assimilation, dans le discours, entre la variable mathématique et la variable informatique conduisent à un quiproquo : alors que l'élève pose une question sur l'objet algorithmique, inscrivant sa réflexion au sein de l'ETA, l'enseignante répond dans le domaine des mathématiques, ramenant le travail dans l'ETM mais en utilisant le vocabulaire de l'ETA. Elle court-circuite la transposition algorithmique qui assure la fibration entre ETA et ETM. Ce quiproquo est accentué par l'écriture même du programme du lutin *cadeau* dans lequel la variable prévue pour recevoir la réponse du joueur est nommée « variable » (fig. 2, i1 et i7). Dans le discours de l'enseignante, l'inconnue x est également désignée par le terme *variable*:

« $x + 3 = 10$. Ça, c'est une équation de niveau 1. Ma variable, dans ce cas-ci, c'est x . »

Elle s'appuie, sans l'expliciter, sur une analogie de structure entre les deux phrases mathématique et informatique (cf. fig. 3).

Écriture mathématique : $x = \text{terme constant à droite} - \text{terme constant à gauche}$

Écriture informatique : 

Figure 3. Phrases mathématique et informatique

Le fait que cette analogie ne soit pas explicitée provoque une agrégation des genèses discursives algorithmiques et mathématiques, provoquant une confusion entre les deux types de travail.

Cette confusion se manifeste, sous une autre forme, dans la phase de dévolution lors d'une interaction entre un élève et l'enseignante à propos de l'initialisation de cette variable informatique.

Félix: Mettre variable à 1000 sert à rien parce que [??..]

Ens.: Heu Oui, il y avait une raison pourquoi qu'on l'a mis à 1000, parce que sinon il générerait des équations... heu ...

Félix: Théoriquement les équations sont forcées d'être entre 2 et 20

Ens.: Attends un peu, attends un peu

Félix : Il a pas une chance aléatoire que le cadeau tombe juste comme ça

Ens.: Dans le fond, on met une condition et on ne veut pas que ça arrive. C'est un peu ça ... 1000, ce serait étonnant que ça arrive.

Figure 4: échange entre un élève et l'enseignante lors de la phase de dévolution

On constate ici que l'ETA personnel de Noémie est encore en construction. Elle n'a pas achevé sa propre genèse discursive ce qui ne lui permet pas d'appréhender la structure itérative globale de l'algorithme. Elle a retenu de son travail préparatoire que le choix de la valeur d'initialisation permet l'entrée dans la boucle, mais cette connaissance reste purement instrumentale. Il lui est donc difficile d'en rendre compte aux élèves ce qui l'amène en position topogénétique basse. Elle laisse ainsi un élève, Félix, développer un constat faux : « *Mettre variable à 1000 sert à rien* ». Elle tente un argument, sans réelle conviction : « *Dans le fond, on met une condition et on ne veut pas que ça arrive.* » Lors de la phase de régulation, elle reconnaît ne plus être capable de justifier cette initialisation à 1000. « *En fait, quand on dit de mettre la variable à 1000, c'est qu'on te donne une condition, mais on ne veut pas que ça arrive. On ne veut pas justement que ça marche tout de suite. Parce qu'on est allé mettre, je ne sais pas si ça va être clair ce que j'essaie de t'expliquer, mais on veut... heu [silence] Je perds mes mots. En fait, on met la condition, mais on ne veut pas que ça arrive* ». Nous observons ici une enseignante aux prises avec un objet dont elle appréhende mal la double référence : la variable peut-être un objet mathématique ou un objet informatique. Elle ne l'a pas tout à fait reconnu comme un objet informatique, c'est-à-dire comme un contenant que l'on doit déclarer pour pouvoir ensuite recueillir une réponse (celle du joueur). Elle n'a pas construit le rôle algorithmique de cette valeur initiale, qui est d'assurer que le programme entre dans la boucle qui suit. Ce sens ne peut en effet se construire que dans l'ETA. La valeur choisie n'a aucun sens dans l'ETM ce qui amène Félix à dire que « *Mettre variable à 1000 sert à rien* ». Noémie, ancrée dans le plan sémiotico-instrumental de l'ETA ne perçoit que confusément le sens algorithmique : « *Dans le fond, on met une condition*

et on ne veut pas que ça arrive ». Elle sait à quoi sert cette valeur, mais sa conception plus mathématique de la variable la met dans l'impossibilité d'avancer des arguments convaincants : « *Je perds mes mots* ». En termes d'idonéité, elle n'a donc d'autre choix que de botter en touche lors de la phase de régulation en espérant qu'une explication émergera du travail conjoint : « *Je vais juste continuer, peut-être que ça va s'éclaircir en cours de route* ».

2e épisode : difficulté à généraliser sans perdre le sens de la résolution algébrique d'une équation

Rappelons que lors de l'entretien *ante*, l'enseignante déclare que son objectif est « *de consolider les apprentissages sur les équations en utilisant la programmation* » ajoutant plus tard qu'il y a « *aussi une généralisation à faire [...] c'est nouveau, là, il ne sera pas facile de générer des mots [...] la généralisation ne va pas être facile à faire, ça fait que c'est pour ça que ça peut bloquer.* »

Lors de la phase de dévolution, l'enseignante circule, intervenant peu dans les groupes. Conformément à la consigne donnée, les élèves se sont attachés à comprendre le programme qui contient la généralisation de la résolution de l'équation.

Élève: On comprend mais on ne sait pas comment l'écrire...

Ens.: Qu'est-ce que tu comprends?

Élève: bein on comprend que dès que le joueur il touche le cadeau, le cadeau il va dire «voici une équation à résoudre". C'est ça ... on met une ...la variable puis le terme constant à droite - le terme constant à gauche. Là pour le terme constant à droite, il choisit un nombre entre deux et vingt et la même affaire pour le terme constant à gauche. Après ça, il nous demande de [dire] le x. En fait il nous demande juste de faire une addition ici. Il demande l'addition.

Ens. : Donc c'est un niveau 1.

Élève : Et après ça, ici, si on a de bon, ça cache, puis ça joue le ... coup de guitare. C'est un coup de guitare.

Ens: C'est un coup de guitare.

Élève : Plus 1 au trésor.

Ens: Exact.

Élève : Je sais pas comment l'écrire

Ens: Bien, t'as compris ...

Figure 5 :

Notons que cet élève émet une description factuelle du programme. En prononçant « *En fait il nous demande juste de faire une addition ici. Il demande l'addition* », il

montre le peu de sens qu'il donne à cette addition, ne la reliant pas à la résolution d'une équation. La réponse de l'enseignante, « *bien, t'as compris* », valide la compréhension du fonctionnement du programme. Elle tente alors d'appuyer sur la chronogenèse en interrompant la phase de dévolution. Elle provoque une régulation en prenant elle-même en charge la généralisation. Prenant une position topogénétique haute d'enseignante de mathématique, elle utilise la métaphore de la balance pour rappeler les règles opératoires de résolution de l'équation $x + 3 = 10$:

Ens [à l'ensemble de la classe] : Quand je te demandais en classe, on utilisait le principe de la balance. Si je te disais $x + 3 = 10$, tu devais faire - 3 de chaque côté du égal. [...] Fais que si on faisait la balance, on enlève 3 d'un côté, on enlève 3 de l'autre côté. Sauf que quand vient le tempsÇa, ça va me donner $x = 7$. Les deux, c'est pareil. Là ici, quand vient le temps de programmer ça, qu'est-ce que je fais pour arriver à ma réponse? J'ai repris ma variable, j'ai pris mon terme constant à droite, et j'enlève celui qui est à gauche, mon terme constant. C'est ça la phrase qui est ici. [...] donc ici, le « - », c'est parce que c'est l'opération inverse de mon addition. Donc, c'est important que je viens de mettre mon opération inverse de façon à ce que je trouve la bonne réponse.

Figure 6:

Dans le cours usuel de mathématiques, l'enseignante suit les instructions curriculaires qui préconisent de donner du sens à la résolution d'équations algébriques par équivalence d'égalités. Or la situation la conduit à amener sur le devant de la scène le recours à l'opération inverse, induite par la transposition informatique de la résolution de l'équation. La fragilité de la fibration entre ETM et ETA amène Noémie à faire à des raccourcis sémiotiques se basant uniquement sur la forme, le *representanem*, de l'instruction d'affectation de la valeur solution de l'équation



Le processus de généralisation qui a présidé la genèse discursive mathématique de cette formule, puis sa transposition dans l'ETA, devient dès lors invisible aux élèves. L'enseignante, quant à elle, se trouve prise dans une contradiction didactique remettant à nouveau en cause l'idonéité anticipée.

Le passage d'une résolution algébrique par manipulation d'égalités équivalentes à une écriture générale est un saut informationnel⁵ que les élèves ne sont pas en mesure de franchir. Aussi, l'enseignante se retrouve-t-elle seule pour assurer l'avancée du temps didactique. Les hésitations de l'enseignante nous laissent à penser qu'elle n'a

⁵ On appelle saut informationnel, un changement de valeur d'une variable didactique à l'intérieur d'une situation susceptible de provoquer un changement de stratégie.

pas perçu ce saut informationnel, ce qui l'amène à justifier par un raccourci technique l'instruction correspondant à la généralisation. Cet épisode est emblématique des difficultés rencontrées par l'enseignante dans la construction des ETM idoines, en particulier lors des sauts informationnels constitutifs de cette étape et des suivantes. Elle peine à amener les élèves à changer de stratégie, tout en conservant le sens de la résolution d'équations algébriques, autrement dit assurer une fibration harmonieuse entre ETM et ETA.

DISCUSSION ET CONCLUSION

Une collaboration théorique fructueuse

L'analyse proposée ici a été menée à travers un dialogue permanent entre les deux cadres théoriques convoqués. Le modèle de l'action conjointe permet de rendre compte de la dynamique de la co-construction du monde didactique entre enseignant et élèves (Schubauer -Leoni et al, 2007) au sein duquel les espaces de travail prennent place. L'analyse de l'action conjointe met au jour l'émergence des objets de médiations au cœur de l'interaction entre l'enseignante et les élèves. L'identification de traits pertinents confère à ces objets un statut qui éclaire les genèses. L'idonéité peut être appréhendée à travers les interprétations de la tâche, le type de rapport attendu/effectif aux objets et la nature des traits pertinents attribués à l'action sur ces objets par les différents acteurs. La collaboration des deux cadres facilite une mise en relation de la dynamique de l'agir en classe avec celle des différents types de travail. Elle permet notamment de mettre au jour la co-construction des savoirs au sein du travail mathématique.

Une idonéité malmenée

La notion d'idonéité cherche à rendre compte de l'adéquation de l'espace de travail proposé par l'enseignant à ces élèves à certaines contraintes qu'il cherche à respecter. Ces contraintes peuvent aussi bien être institutionnelles que de l'ordre des croyances ou des conceptions de l'enseignant. Souvent, elles concernent le travail attendu par les élèves, en lien avec l'ET de référence. Dans la séance que nous avons analysée, l'équilibre habituel de ces contraintes est rompu. L'enseignante cherche à répondre à la fois à l'incitation institutionnelle d'intégrer la programmation informatique et à l'incitation académique (celle du projet de recherche) d'articuler travail algorithmique et travail mathématique. Les contraintes qui en résultent ne sont pas encore les siennes. Elle cherche encore à les apprivoiser et cela la rend très vulnérable. Il est dès lors difficile de parler d'idonéité car l'espace de travail qu'elle propose n'est pas vraiment le sien. Elle est donc conduite à le baliser et à en élémentariser les savoirs par peur de s'y perdre avec ses élèves. Le recours à une tâche construite par un collègue s'avère finalement plus handicapant que sécurisant. C'est un piège dans lequel peuvent tomber facilement les enseignants débutants dans l'usage de la programmation informatique.

Une transposition difficile :

Notre analyse fait ressortir la difficulté de construire des savoirs mathématiques en s'adossant à une situation demandant des connaissances d'ordre informatique et algorithmique. D'une façon générale, travail mathématique et travail algorithmique sont difficiles à articuler. Les fibrations externes sont nécessaires pour réaliser la tâche et peuvent ne pas se mettre en place spontanément. Une fibration shuntée, forcée ou mal assurée peut-être contre-productive en termes de construction du sens. Le passage dans l'ETA force, en effet, une réinterprétation des objets mathématiques. Le langage de programmation et les pratiques algorithmiques amènent un nouvel usage de ces objets. Les généralisations algébriques et algorithmiques ne prennent pas la même forme notamment dans la gestion de la dénomination.

Des enjeux didactiques difficile à identifier

Démêler les enjeux didactiques propres au travail algorithmique s'avère crucial au moment où la tendance est à une vision purement instrumentale de la programmation informatique et de l'algorithmique qui la sous-tend. Dans l'étude présentée ici, l'entretien ante montre que ceux-ci ne sont pas clairement identifiés par l'enseignante : alors que l'objectif premier de cette situation est de provoquer une généralisation mathématique comme support à la programmation d'un jeu, le travail reste essentiellement instrumental. Ce constat recoupe plusieurs recherches montrant combien des objets d'enseignement déclarés à l'interface de deux disciplines sont difficiles à transférer dans leur double référence (Schubauer-Léoni, Leutenegger & Forget, 2007; Devos-Prieur & Grandaty, 2011 ; Couderette, 2016).

REFERENCES

- Barma, S. (2018). *Rapport final : Réaliser une étude de cas multiple qui vise à affiner les connaissances sur l'usage pédagogique ou didactique de la programmation dans les écoles du Québec*. Québec : Université Laval.
- Briant, N. (2013). *Étude didactique de la reprise de l'algèbre par l'introduction de l'algorithmique au niveau de la classe de seconde du lycée français*. Thèse de doctorat. Université Montpellier 2.
- Couderette, M. (2023). Algorithmique au primaire. Premiers résultats d'une recherche en cours. *Actes du colloque de la COPIRELEM « Mathématiques et diversité à l'école. Aider les élèves, accompagner les professeurs »*. Marseille. France.
- Couderette, M. (2016). Enseignement de l'algorithmique en classe de seconde. Une introduction curriculaire problématique. *Annales de Didactique et de Sciences Cognitives*, 21, 267-296. <http://hdl.handle.net/20.500.12162/1908>
- Kuzniak, A., Montoya-Delgadillo, E., et Richard, P. R. (Eds.). (2022). *Mathematical Work in Educational Context: The Perspective of the Theory of Mathematical Working Spaces* (1st ed., Mathematics Education in the Digital Era, Vol. 18). Springer Cham. <https://doi.org/10.1007/978-3-030-90850-8>.

- Kuzniak, A., et Masselin B. (2023). Contrats fortement didactique et ETM idoines. Le cas d'une tâche en probabilités. Dans C. Derouet, A. Nechache, P.R. Richard, L. Vivier, I.M. Gómez-Chacón, A. Kuzniak, M. Maschietto & E. Montoya Delgadillo, *Actes du septième symposium d'Étude sur le Travail Mathématique* (pp. 403-416). IREM de Strasbourg.
- Lagrange, J. B., & Laval, D. (2023). Connecting algorithmics to mathematics learning: a design study of the intermediate value theorem and the bisection algorithm. *Educational Studies in Mathematics*, 112(2), 225-245.
- Masselin, B. (2019). Etude du travail de l'enseignant autour de la simulation en classe de troisième et seconde: métamorphoses d'un problème au fil d'une formation en probabilité (Doctoral dissertation, Université Sorbonne Paris Cité).
- Radford L. (2014) The progressive development of early embodied algebraic thinking. *Mathematics Education Research Journal* 26, 257-277.
- Schubauer -Leoni M.L., Leutenegger F., Ligozat F. et Fluckiger A. (2007). Un modèle de l'action conjointe professeur-élèves : les phénomènes didactiques qu'il peut/doit traiter, In G. Sensevy & A. Mercier (Eds.) (2007), *Agir ensemble. L'action didactique conjointe du professeur et des élèves*. Rennes, Presses Universitaires de Rennes, p. 51-91.
- Sensevy, G. (2007). Des catégories pour décrire et comprendre l'action didactique. In G. Sensevy & A. Mercier (Eds.) (2007), *Agir ensemble. L'action didactique conjointe du professeur et des élèves*. Rennes, Presses Universitaires de Rennes, p. 13-49.
- Venant, F, Jeannotte, D, Passaro, V, Saboya, M, Guillemette, E et E. Knoll. (2022). Pensée algébrique et pensée algorithmique : réflexion autour d'une tâche de généralisation. *Actes du colloque EMF 2022*, Cotonou, Bénin,
- Venant, F., Passaro, V., Jeannotte, D., Knoll, E et M. Saboya. (2023). Articuler travail mathématique et travail arithmétique. *Actes du septième symposium d'Étude sur le Travail Mathématique* (pp. 215-224. IREM de Strasbourg.
- Venant, F. (2018). Programmer les mathématiques: la pensée informatique à l'école primaire. *Bulletin AMQ*, 58(3), 57-70.
- Venant, F. (2022). (2022). Conception de tâches de programmation informatique pour l'enseignement des mathématiques : prendre en compte les enjeux algorithmiques. Abboud, M., & de Hosson, C. (eds). *Actes du colloque «Rendez-vous en didactique: recherches, dialogues et plus si affinités»*.