

At 12 : Initiation à la programmation et enseignement de la géométrie au cycle 3

Maha Abboud, Isabelle Bois et Cécile Kerboul

ESPE de l'Académie de Versailles et IREM de Paris ;

maha.blanchard@u-cergy.fr ; isabelle.bois@u-cergy.fr ; cecile.kerboul@u-cergy.fr

Résumé : Avec les nouveaux programmes, les enseignants du cycle 3 ont vu arriver une nouvelle thématique à travailler avec leurs élèves : l'initiation à la programmation, notamment «à travers la programmation de déplacements ou de construction de figures». Se pose alors la question de l'impact de l'utilisation de logiciels de programmation sur les apprentissages géométriques. Comment faire pour enseigner des notions géométriques avec des logiciels de programmation comme Scratch ? Qu'apporte l'utilisation de tels logiciels aux apprentissages géométriques ? Faut-il changer les pratiques usuelles d'apprentissage de la géométrie, ou bien les activités articulant programmation et géométrie peuvent-elles être isolées du reste et traitées comme visant principalement l'initiation à la programmation ? Nous proposons d'éclairer ces interrogations à partir d'une part, d'un regard critique sur les programmes officiels et d'une étude croisée de quelques manuels et d'autre part, d'observations de classes dont nous présentons une analyse synthétique des résultats.

Mots clés : géométrie ; programmation ; algorithmique ; Scratch ; cycle 3 ; construction de figures ; utilisation de logiciels en classe

Introduction

L'arrivée du thème de la programmation dans les programmes du cycle 3 constitue la concrétisation d'un discours fréquent et divers mis en avant depuis plusieurs années dans la sphère médiatico-politique qui peut se résumer par : « initier les enfants depuis leur plus jeune âge à l'informatique, permet de former des futurs citoyens qui comprennent le monde numérique dans lequel on vit ». Les spécialistes et les chercheurs en informatique ont appuyé également, à travers leurs propres approches, cette demande récurrente en argumentant l'intérêt de l'initiation à la pensée informatique et algorithmique dès l'école primaire : « l'informatique apporte une nouvelle manière de penser et il est important d'initier les élèves à la pensée informatique ou computationnelle » (Dowek, 2016) ; « il y a donc un enjeu fort de culture générale scientifique, de culture générale informatique pour tous [...] Il faut donc initier tous les élèves aux notions de base de l'informatique: algorithmique, langage et programmation » (Abitboul, 2015) ; « il est important que les élèves puis les adultes soient conscients de l'activité humaine à l'origine de toute application informatique, et de la rationalité qui préside à cette activité. Le repérage des difficultés nous conduit à penser qu'il est utile que l'initiation commence tôt de façon que les processus de maturation conceptuelle des élèves puissent avoir lieu sur le temps long des études » (Lagrange & Rogalski, 2015).

Ce nouveau thème est donc apparu à la rentrée 2016 dans les programmes de mathématiques ; les enseignants du cycle 3 ont à l'aborder en articulation avec des domaines du programme de mathématiques, en particulier celui des activités spatiales et géométriques.

Cette nouveauté paraît à même de soulever plusieurs interrogations chez les enseignants de CM1 – CM2 et de 6ème. Comment faire pour enseigner des notions géométriques avec des logiciels de programmation comme Scratch ? Qu'apporte l'utilisation de tels logiciels aux apprentissages géométriques ? Faudra-t-il changer les pratiques usuelles d'apprentissage de la géométrie, ou bien les activités articulant programmation et géométrie peuvent-elles être isolées du reste et traitées comme visant principalement l'initiation à la programmation ?

Notre travail de recherche vise à éclairer ces interrogations et à explorer des pistes de réponses. Dans un premier temps, nous nous fixons comme objectif d'observer les activités que des enseignants de CM2 et de 6ème vont mettre spontanément en place en se basant sur les ressources pédagogiques mises à leur disposition. Nous avons choisi de nous centrer sur des activités liées à la construction de figures géométriques en utilisant le logiciel Scratch. Le second temps de notre travail mettra le focus sur la formation initiale des enseignants et sur la création de ressources au service des formateurs.

Dans cet article, nous rendons compte des résultats de la phase exploratoire de notre recherche. Une première partie, présente notre lecture des programmes et étude des manuels. Les observations de classes sont présentées dans une deuxième partie, suivies d'une analyse synthétique des résultats.

Contexte

Dans cette première partie, nous dressons d'abord un panorama des instructions officielles et des programmes relativement à l'enseignement de la programmation à l'école primaire. Nous étudions ensuite quelques manuels du cycle 3 (CM1, CM2 et 6ème) pour tenter d'y détecter les différentes approches adoptées par les éditeurs et leur éventuelle évolution au cours du cycle.

Les programmes

Au cycle 1¹, les élèves rencontrent les premiers algorithmes pour lesquels il est attendu qu'ils sachent identifier les principes de leurs organisations et poursuivre leurs applications. Ils sont ainsi invités à « organiser des suites d'objets en fonction de critères de formes et de couleurs ; les premiers algorithmes qui leur sont proposés sont simples. Dans les années suivantes, progressivement, ils sont amenés à reconnaître un rythme dans une suite organisée et à continuer cette suite, à inventer des « rythmes » de plus en plus compliqués, à compléter des manques dans une suite organisée » (p. 15).

Cette première familiarisation avec les algorithmes se poursuit à l'école élémentaire et le thème de la programmation vient s'inscrire dans le domaine « Espace et Géométrie ».

Au cycle 2², dans la continuité du cycle 1, les algorithmes concernés restent simples. Les programmes incitent fortement, sans prescrire, à utiliser un logiciel de programmation pour « (se) repérer et (se) déplacer ». C'est la seule compétence travaillée dans ce cycle. « Dès le CE1, les élèves peuvent coder des déplacements à l'aide d'un logiciel de programmation adapté, ce qui les amènera au CE2 à la compréhension, et à la production d'algorithmes simples » dans des espaces familiers, sur un quadrillage, sur un écran (p. 84-86). En parallèle, le document d'accompagnement

¹ *Programme d'enseignement de l'école maternelle*, arrêté du 18-2-2015

² *Programmes pour les cycles 2 3 4*, MENESR, 2016, p 82-87

pour l'évaluation des acquis du SCCC³ apporte, sans être prescriptif, des pistes pour cette évaluation. Par exemple, « l'aptitude à anticiper les différentes étapes d'un déplacement peut être évaluée dans l'écriture de programmes – exprimés en langage courant ou dans un langage spécifique – permettant de produire une succession de déplacements d'un personnage pour se rendre d'un point à un autre dans un environnement donné sur écran ou sur un quadrillage » (p. 5). Ce qui était incitatif au cycle 2 devient une prescription au cycle 3⁴. La programmation occupe une place conséquente dans le programme de mathématiques. Notons toutefois qu'elle est abordée uniquement dans le domaine « Espace et Géométrie ». Les activités géométriques, de repérage ou de déplacement ciblent trois compétences, soit deux de plus qu'au cycle 2 : « programmer les déplacements d'un robot ou ceux d'un personnage sur un écran » ; « reproduire, représenter, construire : des figures simples ou complexes (assemblages de figures simples) » ; « réaliser une figure simple ou une figure composée de figures simples à l'aide d'un logiciel » (p. 210-211). On notera que la première compétence « programmer les déplacements d'un robot ou ceux d'un personnage sur un écran » est présentée comme une situation-activité au cycle 2. Concernant les situations-activités qui l'accompagnent, le programme suggère, sans plus de précision, de « travailler (...) avec de nouvelles ressources comme les systèmes d'information géographique, des logiciels d'initiation à la programmation » (p. 8). Concernant les deux autres compétences, aucun exemple de situations-activités n'est donné si bien que, par exemple, pour atteindre la compétence « réaliser une figure simple ou une figure composée de figures simples à l'aide de logiciel », on ne sait quel(s) logiciel(s) est (sont) préconisé(s) (logiciels de géométrie dynamique ? logiciels de programmation ? autres ?). Par ailleurs, le document d'accompagnement pour l'évaluation des acquis du SCCC⁵ précise que celle-ci porte sur « l'aptitude à lire ou à écrire un programme, dans un langage approprié, permettant l'exécution d'un déplacement d'un robot ou d'un personnage sur un écran » et plus seulement, comme au cycle 2, sur son anticipation » (p. 8). Cette étude des programmes et des documents d'accompagnement laisse entrevoir quantité de zones d'ombre laissées à la charge de l'enseignant. L'étude de manuels apporterait-elle une sorte de traduction des programmes pour accompagner l'action de l'enseignant ?

Les manuels

Les manuels scolaires du cycle 3 font des choix très différents par rapport à l'application des programmes relatifs à la programmation et à l'intégration des technologies dans l'enseignement de la géométrie. Les approches choisies relativement aux trois niveaux (CM1, CM2 et 6ème) au sein d'une même collection se révèlent parfois être non cohérentes. Nous ne relatons pas ici une étude exhaustive de l'ensemble des manuels, mais plutôt une vision généraliste à travers l'étude de quelques-uns⁶. Nous nous intéressons particulièrement à la manière dont la programmation est prise en compte dans les leçons et les exercices et à la place des logiciels de programmation dans la palette des outils numériques présentés pour le travail de l'élève.

Manuels de CM1 et CM2

³ Document d'accompagnement pour l'évaluation des acquis du socle commun de connaissances, de compétences et de culture. Éléments pour l'appréciation du niveau de maîtrise satisfaisant en fin de cycle 2 (2016)

⁴ Programmes pour les cycles 2 3 4, MENESR, 2016, p 209-214

⁵ Document d'accompagnement pour l'évaluation des acquis du socle commun de connaissances, de compétences et de culture. Éléments pour l'appréciation du niveau de maîtrise satisfaisant en fin de cycle 3 (2016)

⁶ Il s'agit des manuels dont nous disposons au début de l'étude.

Les manuels de CM1 et de CM2 ne traitent pas tous de la programmation et font, généralement, d'autres choix comme celui d'utiliser des logiciels de géométrie dynamique, le tableur, la calculatrice (lesquels figurent dans les anciens programmes de 2008). Quand ils l'abordent, ce peut être sous la forme d'une « méthode » en vue de programmer un déplacement ; la programmation au service de la réalisation d'une figure géométrique se trouve plus fréquemment dans les manuels de CM2.

Arrêtons-nous sur les manuels de CM1 (2016) et de CM2 (2017) de deux collections : *Graine de maths*, Nathan et *Les nouveaux outils pour les maths*, Magnard. Dans *Graine de maths CM1 (CM2)*, Nathan, 2016 (2017), deux logiciels de programmation sont utilisés : Géotortue et Scratch. Ils sont vus comme des « logiciels de géométrie » associés à la programmation des déplacements en CM1 (p. 136-137) et en CM2 (p. 146-147).

• Géotortue et Scratch sont deux logiciels de géométrie qui permettent de programmer des déplacements.

Action	Instruction Géotortue	Instruction Scratch
Avancer de 100 pas.	av 100	
Tourner à droite d'un angle droit.	td 90	
Tourner à gauche d'un angle droit.	tg 90	

Voici le déplacement de la tortue de Géotortue obtenu avec ce programme.

```
> av 100
> td 90
> av 100
> tg 90
> av 25
```



Voici le déplacement du chat de Scratch obtenu avec ce programme.




Figure 1 : Extrait de *Graine de maths* CM1, Nathan (2016), p. 136

Entre le CM1 et le CM2, la vision évolue. En CM2, deux parties apparaissent avec une nette distinction entre *réaliser* et *construire* une figure géométrique : premièrement, seul Scratch (mais pas Géotortue) est associé à la nouvelle partie « réaliser une figure simple à l'aide d'un logiciel de programmation » (p 146). Deuxièmement, le logiciel de géométrie dynamique GeoGebra, absent dans le manuel de CM1, fait son entrée ; il permet la construction de figures (p. 178).

Quant à la collection *Les nouveaux outils pour les maths CM1 (CM2)*, Magnard, 2016 (2017), les visions relatives à la programmation et à l'utilisation d'un logiciel de géométrie dynamique (GeoGebra) sont identiques en CM1 et en CM2 : le logiciel Scratch sert à animer un lutin et il n'y a aucun lien entre le logiciel de programmation et la programmation d'un déplacement ou de la réalisation d'une figure (CM2, p. 148-149).

Je retiens

- Scratch est un logiciel qui sert à écrire des scripts (petits programmes) pour animer un lutin (personnage ou objet).
- Le script est composé de blocs de commande assemblés dans un ordre précis. Il débute par la commande qui déclenche l'action.
- Pour faciliter l'écriture des scripts, on peut utiliser des commandes « boucles » et des commandes « conditions ».

Commande qui déclenche l'action :



Commande boucles



Commande conditions



Figure 2 : Extrait *Les nouveaux outils pour les maths*, CM2, Magnard, 2017, p. 148

Graine de Maths et *Les nouveaux outils pour les maths* adoptent ainsi deux visions différentes de la programmation. Ce qui différencie également les deux manuels est la manière d'intégrer la

géométrie dynamique : *Les nouveaux outils pour les maths* propose d'utiliser GeoGebra en CM1 et en CM2, alors que *Graine de maths* ne le propose qu'en CM2 (p.152-153). Par contre, les visions de ce logiciel de géométrie se rejoignent dans les deux manuels : permettre la construction de figures.

Manuels de 6^{ème}

Sur un panel de sept manuels de 6^{ème} parus en 2016 ou 2017, nous observons que le logiciel de géométrie dynamique GeoGebra et le tableur y trouvent leur place (dans les programmes depuis plusieurs années). Nous ne pouvons cependant en dire autant concernant Scratch. Ainsi certaines collections l'ont peu - voire pas - intégré (*Kwyk 6^e*, Hachette, 2016 ; *Transmath 6^e*, Nathan, 2016) alors que d'autres ont essayé de l'aborder dans la plupart des chapitres (*Mission Indigo 6^e*, Hachette, 2017, *Maths Monde 6^e*, Didier, 2017).

On peut y trouver des tentatives pour définir Scratch auprès des élèves. Par exemple :

- “un **logiciel** qui permet de faire exécuter des commandes à un ou plusieurs lutins” (*Mission Indigo 6^e*, Hachette, 2017, p 10) ou encore « un **logiciel** qui permet de programmer en langage naturel » (*Phare 6^e*, Hachette, 2017, livret d'accompagnement p. IX)
- « un **langage** de programmation visuel » (*Maths Monde 6^e*, Didier, 2017, p. 260)

Enfin, les activités proposées sont de natures très diverses : certaines sans lien avec les mathématiques, d'autres dans le cadre numérique ou géométrique.

En ce qui concerne les activités autour de la construction de figures avec Scratch, on retrouve une structure assez commune dans plusieurs manuels et conforme aux programmes : un script étant proposé, il s'agit dans un premier temps, soit de l'analyser (en faisant par exemple un dessin à main levée), soit de l'exécuter directement à l'aide du logiciel pour voir la figure géométrique tracée par le lutin. Puis, dans un second temps, il est demandé de modifier ce script (ou de l'adapter) pour répondre à une consigne donnée.

Figure 3 : Extrait de *Maths Monde 6^e*, Didier, 2017, p. 192

Expérimentations

A l'issue de ce premier travail d'étude des ressources disponibles, programmes et manuels, nous nous sommes dirigées vers les observations de classe autour de l'utilisation du logiciel Scratch. Des discussions avec plusieurs enseignants nous ont révélé que face à la variabilité dans les manuels, et pour se conformer aux programmes, ces enseignants se dirigent souvent vers les documents

d'accompagnement et les ressources proposées sur les sites officiels, en l'occurrence ici le document : « Initiation à la programmation aux cycles 2 et 3 »⁷.

Nous avons proposé à une enseignante de classe de 6^{ème} (Cindy) de mettre en place une séance utilisant le logiciel Scratch en se basant si possible sur les documents d'accompagnement des programmes. Elle s'est emparée de la ressource « Initiation à la programmation. Annexe 5.4 : Scratch – Figures Géométriques »⁸ sur le site Eduscol. Elle l'a cependant adaptée afin de donner aux élèves l'opportunité de produire par eux même un script, sans « modèle ». De plus, elle espérait profiter d'une telle situation pour introduire les notions de répétitions et de boucles (cf. annexe 1).

Elle a mis en place cette activité après une séance en débranché sur les déplacements et deux séances de familiarisation avec Scratch en salle informatique. Les 30 élèves de la classe ont travaillé par binômes (un binôme par ordinateur). Le travail de trois binômes a été observé et retranscrit. L'enseignante n'a pas fait d'interventions collectives pendant la séance, elle a cependant demandé aux élèves d'enregistrer régulièrement leurs scripts pour une exploitation ultérieure. Nous avons eu accès à ces scripts.

L'activité conçue par Cindy a été ensuite proposée à Gilles, un enseignant en double niveau CP/CM2. Il a décidé de la mettre en place avec ses 10 élèves de CM2. En début de séance, l'enseignant a réuni les élèves autour d'une table et a organisé un bilan collectif sur les activités déjà faites avec Scratch. Ces interventions collectives ont été filmées et transcrites. Par la suite, les élèves ont travaillé individuellement sur ordinateur. Le travail de trois élèves a été observé et retranscrit.

Dans cet article, nous ne détaillons pas les analyses faites pour chacune des deux séances. Nous optons plutôt pour la présentation des résultats d'une façon synthétique en les illustrant par des extraits des observations.

Résultats

Un regard synthétique sur les résultats partant de notre problématique initiale nous permet de les regrouper selon trois entrées.

Mobiliser des connaissances mathématiques pas encore là !

L'angle de 90°

Le logiciel Scratch propose les commandes « tourner à droite de ... degrés » et « tourner à gauche de ... degrés ». Or d'après les instructions officielles, l'introduction d'une unité de mesure d'un angle et l'utilisation d'un outil de mesure ne sont abordées qu'à partir du collège.

Pour esquiver l'utilisation de cette notion, beaucoup de manuels de CM1 et CM2 ne proposent que des activités de déplacement et/ou de construction géométrique avec Scratch n'utilisant que des blocs préformatés à 90° (même en 6^{ème}). Certains manuels s'aventurent cependant un peu plus loin : par exemple, dans *Les nouveaux outils pour les maths CM2* (Magnard, 2017, n°5 p 149) où l'angle

⁷http://cache.media.eduscol.education.fr/file/Initiation_a_la_programmation/92/6/RA16_C2_C3_MATH_initiation_programmation_doc_maitre_624926.pdf

⁸http://cache.media.education.gouv.fr/file/Initiation_a_la_programmation/91/6/RA16_C2_C3_MATH_annexe_5_4_scratch_figures_geo_624916.pdf

de 45° est présenté comme un demi-angle droit (comme un code à connaître) ou encore dans *Maths Monde 6^{ème}* (cf. Fig. 3) avec la construction d'un triangle équilatéral.

Cindy et Gilles sont bien conscients de cette ambiguïté, passée sous silence lorsqu'il s'agit d'activités avec Scratch.

Gilles prend en charge la notion de mesure d'angle lors du regroupement en début de séance. Il introduit l'unité de mesure d'un angle et l'instrument de mesure (il montre un rapporteur de tableau aux élèves) et note sur un affichage qu'un angle droit mesure 90° . Quant à Cindy, elle explique, lors d'un entretien post-observation, qu'elle n'a pas encore travaillé sur les angles à l'heure de l'expérimentation Scratch. Elle a cependant anticipé ce problème et profité de l'activité en débranché sur les déplacements pour montrer aux élèves que « tourner à droite ou à gauche » ne suffit pas pour guider une personne et qu'il faut donc être plus précis : « tourner comment et de combien ? ».

L'observation des élèves montre qu'ils insèrent quasi-systématiquement « 90 » dans le bloc « tourner de ... degrés », reprenant ainsi les instructions préalables de l'enseignant. S'ils écrivent un autre nombre, ils corrigent rapidement en le remplaçant par 90. A l'évidence, ils manipulent ce « 90 », tel « un code », sans réellement comprendre la notion d'unité de mesure d'angle sous-jacente.

Quelques élèves « bricolent » avec ces angles de 90° pour obtenir l'orientation souhaitée. Par exemple, lorsqu'ils veulent remettre à l'endroit leur lutin qui a « la tête en bas », ils répètent deux fois « tourner de 90 degrés » (avec ou sans boucle) pour faire un demi-tour (cf. annexe 2).

Le repère du plan et les coordonnées des points

Lorsqu'on utilise le logiciel Scratch, on peut être amené à travailler sur la notion de coordonnées dans le plan. La scène est en effet munie d'un repère (visible ou pas, au choix de l'utilisateur), qui permet notamment de définir la position du lutin à un instant t. Or, d'après les instructions officielles, se repérer dans le plan muni d'un repère orthogonal est au programme du cycle 4.

La notion de coordonnées est utilisée cependant dans certains manuels, comme par exemple dans exercices de programmation du manuel *Les nouveaux outils pour les maths*, CM2, Magnard, 2017 (p. 146-147). Pour les auteurs de ce manuel, il semble qu'il soit nécessaire, vraisemblablement pour garder une certaine cohérence, d'introduire la notion de coordonnées ou de rappeler la notion des points cardinaux en amont de la programmation (dans la partie « Se repérer et se déplacer dans l'espace »).

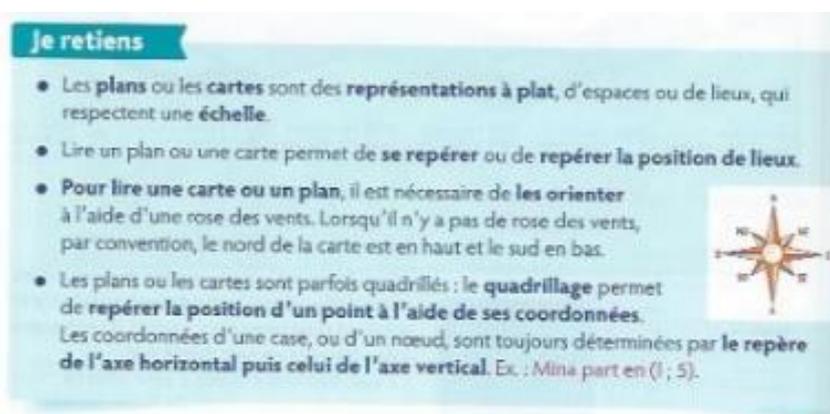


Figure 4 : Extrait *Les nouveaux outils pour les maths*, CM2, Magnard, 2017, p 146

Dans le manuel *Mission Indigo 6^e* (Hachette, 2017, p 144), les auteurs ne cherchent pas à définir ce que sont les coordonnées d'un point (puisque ce n'est plus au programme). Il est cependant utile parfois de commencer le script au centre de la scène et c'est à cette occasion que la commande « aller à x:0 et y:0 » est introduite. A noter que pour les élèves, ceci n'a pas de sens mathématique, d'autant plus que le repère n'est pas visible sur la scène.



Figure 5 : Extrait *Mission Indigo 6^e*, Hachette, 2017 (p 144)

Nous présentons ci-après quelques observations de classe relativement à cette entrée.

En début de séance de CM2, lorsque l'enseignant procède à un rappel des séances précédentes, il demande aux élèves de rappeler quelle est la différence entre les commandes « tourner » et « s'orienter », ce qui l'amène à revenir sur la notion de coordonnées. Après ce rappel et lorsqu'il passe aider individuellement les élèves, nous remarquons que l'enseignant a besoin des coordonnées, en particulier pour réinitialiser son programme, et qu'il couple alors le bloc « aller à » au bloc « s'orienter à 90 ». Pour un des élèves observés, nous remarquons que, malgré l'aide de l'enseignant pour orienter le chat, il ne semble pas avoir conscience de ce qu'il fait ; pour lui, les blocs « tourner » et « s'orienter » restent confus.

Le script proposé ci-dessous laisse supposer que la commande « aller à... » est souvent utilisée dans son sens commun, c'est-à-dire pour faire aller le lutin d'un point à un autre point, lesquels étant repérés par des coordonnées visibles à l'écran. Ce script est produit par un élève de CM2 après 7 minutes de recherche de la question 1. Auparavant, il écrit à l'issue d'une seconde tentative le bloc de réinitialisation. Il le teste en demandant au lutin « avancer 100 ». L'élève cherche alors à faire faire un quart de tour au chat. Il utilise en première intention la commande « aller à... ».

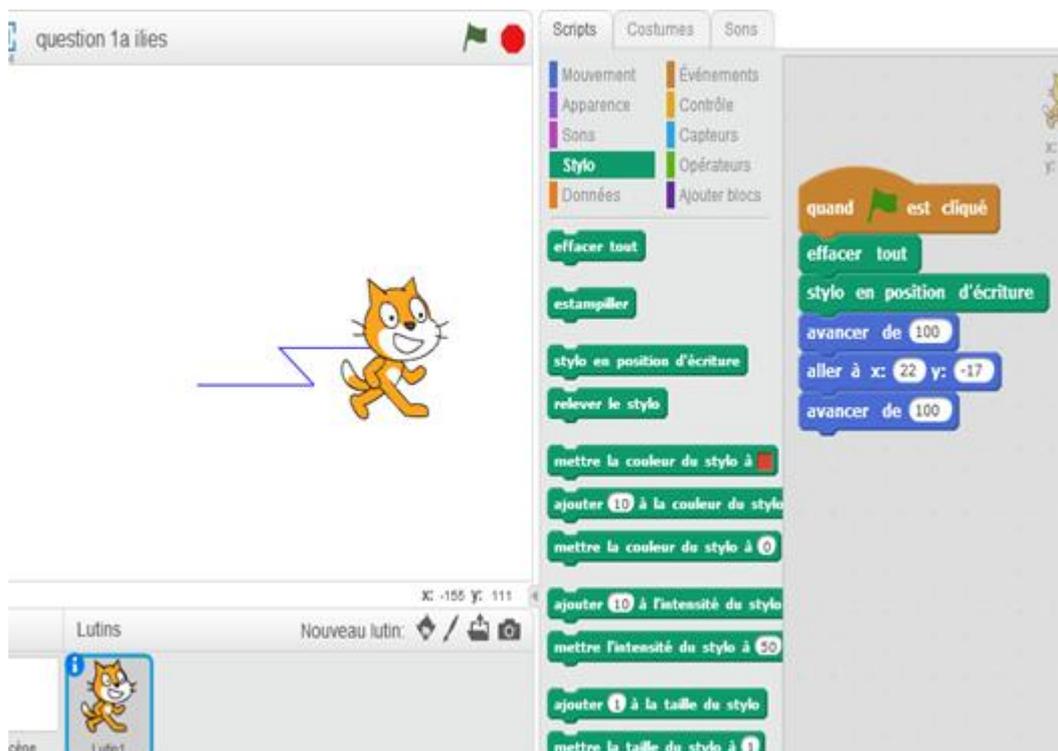


Figure. 6 : Script d'Iliès, tentative de réponse à la question 1

Nous avons également observé une autre utilisation contrainte des coordonnées (toujours avec la commande « aller à... ») pour replacer, dans le champ de la scène, le lutin qui sort de l'écran. Certains élèves peuvent donc recourir aux coordonnées par nécessité relative à la manipulation, sans pour autant convoquer les connaissances mathématiques sous-jacentes. Cette observation permet par ailleurs de pointer une des difficultés techniques d'utilisation du logiciel Scratch liées au comportement du lutin aux bords de la scène visible à l'écran et liées également à son interprétation par l'élève.

Nous pouvons conclure ici que l'utilisation de Scratch amène les élèves de cycle 3 à mobiliser des connaissances qui relèvent du programme de cycle 4 et non de leur cycle d'enseignement. Ils s'en sortent (presque) toujours en recourant à des stratégies d'essais-erreurs ou en demandant de l'aide à l'enseignant jusqu'à comprendre les fonctionnalités de certaines commandes. Cette compréhension semble rester à un niveau empirique, mais y a-t-il un « sens mathématique » qui vient s'y adjoindre, ou déclenche-t-elle la recherche d'un tel sens ?

Construction de figures géométriques : oui mais !

Les programmes et les documents d'accompagnement soulignent que le logiciel Scratch permet de tracer des figures géométriques et certains manuels proposent des activités allant dans ce sens. Toutefois nos observations nous amènent à poser trois grandes questions relatives à ce type d'activités :

Les élèves construisent-ils des figures géométriques ou bien s'agit-il pour eux de « dessiner » des chemins, si possible fermés ?

En effet, nous observons souvent les élèves répéter : « avancer de x », « tourner de 90 » jusqu'à ce que le dessin se ferme, mais sans vraisemblablement tenir compte du fait qu'avancer correspond à « construire un segment » ou bien que tourner de 90 correspond à « préciser la mesure d'un angle ».

Nous donnons ici l'exemple d'un élève de CM2 qui écrit un script correct pour dessiner la figure de la question 1 et appelle l'enseignant pour la validation. L'enseignant lui pose la question si c'est « toujours un carré » et l'élève répond « non ». S'ensuit un dialogue où l'enseignant essaye de faire dire à l'élève les propriétés des carrés, qui sont bien vérifiées par le script, pour qu'il se rende compte qu'il a vraiment construit un carré. Ce sont donc uniquement les questions de l'enseignant qui guident vers la mobilisation de connaissances géométriques : formuler l'égalité des angles et reconnaître l'égalité des angles.

Assiste-t-on à une régression dans la progression de l'apprentissage de la géométrie ?

Rappelons que les recherches didactiques ont montré un cheminement dans l'apprentissage qui part de la géométrie perceptive, évoluant vers la géométrie de l'argumentation basée sur l'utilisation des propriétés et vers enfin la géométrie déductive. En particulier, Houdement et Kuzniak (2006) montrent la progression (du cycle 1 au cycle 4) de l'apprentissage de la géométrie (GI) qui a pour source de validation la réalité, le sensible et qui est basée sur l'expérience, vers la géométrie (GII) axiomatique naturelle où la source de validation se fonde sur les lois hypothético-déductives dans un système axiomatique aussi précis que possible. Les exemples que nous avons observés ne montrent-ils pas un risque de perturbation de cette progression ? Assiste-t-on à un mouvement, inverse, de la GII vers la GI qui d'ailleurs n'est pas présent, rappelons-le, lors de l'utilisation de logiciels de géométrie dynamique ?

Quelle appréhension ont les élèves de la figure géométrique lorsqu'ils travaillent avec Scratch ?

Lors du travail sur des figures géométriques, les élèves peuvent mobiliser différentes appréhensions de la figure (Duval, 1994). Lors de la reproduction de figures avec Scratch, les élèves en ont nécessairement une appréhension perceptive, cependant la discrimination en sous-figures ne coïncide pas nécessairement avec les unités figurales prises en compte dans la construction de la figure modèle. De même, l'ordre de prise en compte des unités figurales qui entrent dans la construction d'une figure (appréhension séquentielle) et qui dépend principalement des propriétés mathématiques à représenter et des constructions techniques des instruments utilisés (logiciels de géométrie dynamique, règle et compas...) n'est plus le même.



Citons ici l'exemple de la question 3 de l'annexe 1 : plusieurs élèves ont vu ici un grand rectangle qu'il faut d'abord dessiner puis partager par des traits verticaux. Cette stratégie est pertinente, voire économique, dans un environnement papier-crayon. Cependant, la mettre en place dans cette question les aurait obligés à modifier leurs scripts ou à le refaire pour réaliser une frise plus longue.

On voit ainsi des confusions multiples entre des procédures et des connaissances mobilisées en papier crayon qui ne sont plus les mêmes avec Scratch et qui viendraient parasiter le travail dans l'un ou l'autre des deux environnements. Ceci est d'autant plus vrai lorsque le projet didactique lié aux connaissances informatiques à mobiliser (la notion de boucle par exemple) n'est pas visible aux élèves.

Connaissances et stratégies relevant de la programmation : approche assumée ou non !

Les boucles

Pour la tâche proposée, le recours aux boucles peut être motivé par le caractère répétitif des instructions, notamment à partir de la question 3. Les élèves observés les ont-ils utilisées ?

Dans la classe de 6^{ème} de Cindy, aucun élève n'a eu, pour première intention, l'idée d'utiliser la boucle « répéter ». Quant aux élèves de CM2 de Gilles, ils l'ont davantage utilisée, vraisemblablement parce que l'enseignant l'avait introduite collectivement au cours d'une séance précédente.

En analysant les scripts, on remarque que plusieurs élèves ont réussi la construction de la frise sans utiliser de boucle : les scripts sont très longs, mais cela ne semble pas les perturber. Dans d'autres scripts, on note que les boucles apparaissent là où l'enseignant ne les attendait pas : par exemple, dans son script (cf. annexe 2), Clément a utilisé une boucle itérative pour permettre au lutin de faire un demi-tour, mais n'a pas eu l'idée d'en utiliser une pour construire son carré de départ.

Nous en concluons que l'utilisation de boucles ne semble pas « naturelle » pour les élèves, même si les contraintes de la tâche sont choisies pour motiver cette utilisation. Un apprentissage plus spécifique de cette notion importante dans le domaine de la programmation serait donc nécessaire.

Les stratégies par bricolage

On observe que les élèves bricolent dans le but de contourner leur(s) difficulté(s) du moment. Ils mettent en œuvre des stratégies personnelles pour pouvoir avancer dans la tâche. Reprenons ici par exemple le moment au cours duquel il s'avère nécessaire de réorienter le lutin. En étudiant les scripts d'Iliès et de Clément (annexe 2), on s'aperçoit que ces deux élèves, n'ayant probablement pas compris la commande « s'orienter à », ne l'utilisent pas. Mais cela ne les empêche pas de progresser dans la tâche. En effet, tous deux bricolent différemment en s'appuyant sur ce qu'ils voient à l'écran : Iliès utilise deux fois de suite la commande « tourner de 90 degrés » tandis que Clément l'utilise une seule fois dans une boucle de deux répétitions.

Par ailleurs, nous avons également montré comment certains élèves parviennent à faire abstraction des notions mathématiques de coordonnées et de repères, par exemple en utilisant la commande « aller à » qui permet au lutin de se déplacer en ligne droite d'un point à un autre après avoir repéré qu'ils étaient associés à des valeurs (sans savoir qu'il s'agit de coordonnées).

Nos observations de classe nous amènent donc à penser que les élèves sont capables de s'approprier ce qu'ils voient à l'écran ou trouvent dans les menus, sans pour autant comprendre les contenus mathématiques sous-jacents. D'ailleurs ce type de stratégies est fréquent lors de l'appropriation d'un artefact technique et les stratégies par essais-erreurs font partie du paysage de la programmation. Mais comment allier ce bricolage à visée de « réussite technique » de la tâche (un programme qui tourne) et le nécessaire recours à un raisonnement fondé sur des connaissances lorsqu'on travaille en classe de géométrie ?

Discussion

Un regard global sur les premiers résultats de notre recherche nous permet d'avancer que l'utilisation d'un logiciel de programmation (ici Scratch) pour faire de la géométrie ne paraît pas compatible avec une intégration « naturelle » dans l'activité géométrique des élèves du cycle 3.

Cette utilisation peut parfois constituer un obstacle à la mobilisation de connaissances géométriques, pourtant supposées acquises, en centrant l'activité sur les connaissances techniques liées à l'environnement. Elle peut aussi venir parasiter les stratégies usuelles d'analyse géométrique de la figure dans le but de la reproduire en introduisant des contraintes obligeant l'élève à penser les

figures comme des lignes continues et non en les déconstruisant en des segments/droites liés par des propriétés géométriques.

Toutefois, une initiation préalable au logiciel dans des séances dédiées (sans avoir l'objectif de « faire des mathématiques »), pourrait décharger l'élève (et aussi l'enseignant) lors des séances mathématiques de la gestion de problèmes liés à des connaissances techniques et centrerait son attention sur une reconnaissance d'un environnement géométrique dans lequel ses connaissances anciennes peuvent être mises à profit. Ceci ouvrirait la palette des tâches géométriques en introduisant une nouvelle nuance autour des régularités des figures et la construction à moindre coût de frises (voire de pavages).

Cette initiation préalable à l'utilisation du logiciel en tant que tel recentrerait l'objectif de cette utilisation sur sa visée originelle, à savoir « l'initiation des élèves à la pensée informatique ». Enseigner cette pensée tendra à développer des compétences adéquates : savoir décomposer un problème en sous-problèmes plus simples ; savoir réfléchir aux tâches à accomplir en termes d'étapes et d'actions (Tchoukinine, 2017) et à adopter une approche orientée « informatique créative » autorisant voire encourageant les stratégies par essais-erreurs et les démarches de projet. Autrement dit, engager l'enseignant de cycle 3 dans un nouveau domaine d'enseignement comme c'est le cas pour le cycle 4, clarifiera, à notre avis, l'enjeu pour l'enseignant et permettra d'éviter des contradictions, parfois handicapantes, entre une approche mathématique « familière » et une approche informatique nouvelle qui n'a pas encore fait ses preuves pour ce niveau d'enseignement.

Cette dernière réflexion pose d'emblée la question de la formation des enseignants. Nous reprenons ici à notre compte des questions avancées par Tchounikine (2017) qui peuvent structurer une telle formation : quels objectifs pédagogiques (des conceptions-approches orientées « algorithmique » et « informatique créative ») peut-on considérer pour l'enseignement de l'informatique ? Que veut dire « utiliser Scratch pour enseigner la pensée informatique » ? Que faut-il comprendre de Scratch en tant qu'enseignant, et comment s'y prendre ? Comment définir et gérer des situations pédagogiques ? Quelles difficultés peut-on attendre et anticiper ?

C'est à construire de telles formations et à les expérimenter que nous consacrons la suite de notre travail de recherche.

Références

Abitboul, S. (2015). L'école au cœur du numérique - enseigner l'informatique.

<http://peep.asso.fr/thematiques/numerique-a-l-ecole/la-peep-l-informatique-et-l-ecole/l-ecole-au-c-ur-du-numerique-enseigner-l-informatique-table-ronde-du-93eme-congres-peep/>. Consulté le 25 septembre 2017

Dowek, G. (2016). En informatique, il est important de commencer tôt.

<http://www.cafepedagogique.net/lexpresso/Pages/2016/06/03062016Article636005337946309506.aspx>. Consulté le 22 septembre 2017.

Duval, R. (1994). Les différents fonctionnements d'une figure dans une démarche géométrique. *Repères IREM*, 17, 121-138.

Houdement, C. & Kuzniak, A. (2006) Paradigmes géométriques et enseignement de la géométrie. *Annales de didactique et de sciences cognitives*, 11, 175-193.

Lagrange J.B. & Rogalski, J. (2015) Savoirs, concepts et situations dans les premiers apprentissages en programmation et en algorithmique. In A.C. Mathé & E. Mounier (Éds.), *Actes du Séminaire National de Didactique des Mathématiques* (pp. 155176). IREM Paris.

Tchounikine, P. (2017). Initier les élèves à la pensée informatique et à la programmation avec Scratch - <http://lig-membres.imag.fr/tchounikine/PenseeInformatiqueEcole.html>. Consulté le 24 septembre 2017.

Annexe 1 : activité de Cindy

Des carrés et des rectangles avec Scratch

1. Le chat veut tracer un carré de longueur 100 pixels.

Créer un script qui permet de tracer un tel carré.

Appeler le professeur.

Enregistrer le script.



2. Le chat souhaiterait maintenant accoler un rectangle de largeur 50 pixels et de longueur 100 pixels au carré qu'il a dessiné précédemment.

Compléter le script précédent.

Appeler le professeur.

Enregistrer le script.



3. Le chat aime beaucoup dessiner !



A partir du script précédent, aide le chat à dessiner la frise ci-dessus.

Appeler le professeur. Enregistrer le script.

