



LaRAC

UGA
Université
Grenoble Alpes

Evaluer l'efficacité de dispositifs éducatifs : méthodes et questions

—
Pascal Bressoux

Séminaire de recherche SFERE, Université Aix-Marseille,
14 février 2024

Cas de l'évaluation de l'efficacité de méthodes et dispositifs pédagogiques

L'efficacité est relative : d'une certaine façon, toutes les méthodes « marchent » (i.e. les élèves progressent)

- La question est de savoir si certaines marchent **mieux** que d'autres
- => comparaison A/B

- Intérêt des essais randomisés contrôlés (RCT*)
- Constitution de groupe expérimental et groupe contrôle
- Randomisation : affectation aléatoire des individus entre les conditions expérimentales

* *Random controlled trials*

- Meilleure manière de s'assurer que **toutes** les caractéristiques (**observées et inobservées**) sont les mêmes entre les groupes
- Les groupes ne varient **que** sur la variable de traitement

- Les variations **d'une** variable, et **d'une seule**, entraînent-elles des variations systématiques dans le phénomène étudié ?
- => raisonnement « toutes choses égales par ailleurs » (strict)

- Ne signifie pas que les RCT ont l'exclusivité de la production de connaissances
- Randomisation pas toujours possible
- Objet pas toujours adapté à l'expérimentation

Un exemple de RCT : L'expérimentation EXPIRE



Laurent, M., Crisci, R., Bressoux, P., Chaachoua, H., Nurra, C., de Vries, E., & Tchounikine, P. (2022). Impact of programming on primary mathematics learning. *Learning and Instruction*, 82, <https://doi.org/10.1016/j.learninstruc.2022.101667>

Equipe pluridisciplinaire composée de chercheurs en

- informatique (LIG-METAH, UGA)
- didactique des mathématiques (LIG-METAH, UGA)
- sciences de l'éducation (LaRAC, UGA)

Collaboration avec la DSDEN de l'Isère

Appel à projets Programme d'Investissements d'Avenir

E-fran



Action « Innovation numérique pour l'excellence
éducative »

Contexte et fondements théoriques

L'introduction du numérique à l'Ecole

- ne se limite pas à la fourniture de matériel (ordinateurs, tablettes, etc.)

- mais vise, de façon plus fondamentale, à **initier les élèves à la « pensée informatique »**

(i.e., savoir appréhender un problème et sa solution à différents niveaux, réfléchir aux tâches à accomplir sous forme d'une série d'étapes – algorithmes –, décomposer un problème en plusieurs problèmes simples et réutiliser ou généraliser des problèmes et des solutions)

- Enjeu important de l'éducation au numérique
=> mettre les élèves en situation de **production**
(construction d'algorithmes et de programmes) **et non**
simplement de consommation

- **Au secondaire** : enseignement numérique souvent introduit comme discipline propre des sciences informatiques (cf. la récente création du CAPES « numérique et sciences informatiques »)
- **Au primaire** : généralement introduit comme support à l'acquisition de disciplines autres, en particulier les mathématiques

Pensée informatique et apprentissages

- pensée informatique vue comme véhicule à l'acquisition de notions mathématiques, voire comme pouvant faciliter leur acquisition

- => pensée informatique considérée comme une compétence intellectuelle générale
- => devrait être facilement transférable à d'autres domaines

“ The educational benefits of being able to think computationally—starting with the use of abstractions—enhance and reinforce intellectual skills, and thus can be transferred to any domain ” (Wing, 2011, p. 7).

- La question du transfert est donc centrale (et semble a priori réglée)
- De plus, spécificité des mathématiques : le travail sur plusieurs registres pourrait aider à la compréhension profonde des concepts mathématiques (Duval, 1999, 2000)
=> registre graphique, algébrique, géométrique, **algorithmique...**

Etat de l'art : quelles preuves de bénéfices pour les apprentissages ?

- Les preuves à l'appui sont peu nombreuses

- Hypothèse déjà à l'œuvre dans le constructionisme de Papert et le développement du langage Logo (1980)
- Mais problèmes liés à la complexité de la syntaxe de programmation
- => intérêt des logiciels de programmation visuelle

- Beaucoup de recherches non convaincantes (e.g., faibles effectifs, pas de randomisation, pas de groupe contrôle, pas de mesures d'acquisitions...)

Une méta-analyse récente a montré des possibilités de transfert positif de la programmation à des situations impliquant des compétences mathématiques (Scherer et al, 2019).

Cependant, les activités de programmation qui y sont décrites sont très diverses (e.g. Lego Robotique), certaines sont obsolètes (e.g. Logo, Basic) et les compétences mathématiques en jeu sont très variées comme la résolution de problèmes, la modélisation, la réussite en général (mesurée par les notes obtenues aux cours) et les connaissances conceptuelles.

Il y a donc un **manque de recherches empiriques récentes** sur le transfert d'apprentissage de la programmation à des concepts spécifiques en mathématiques (Drot-Delange et al., 2019 ; Scherer, 2016 ; Shute et al., 2017).

- L'étude la plus convaincante/rigoureuse est *ScratchMaths* (Benton et al., 2018; Boylan et al., 2018). RCT avec intervention sur deux ans sur 6232 élèves (9-11 ans) en Angleterre. Etude des effets de la programmation avec le logiciel *Scratch* sur :
 - la pensée informatique
 - les compétences mathématiques.
- Les résultats montrent un très faible effet positif sur la pensée informatique (0.10 SD) et aucun effet sur les compétences mathématiques (0.03 SD).
- Mais enseignement non exclusif de *Scratch* (i.e. le groupe expérimental fait des activités *Scratch* parmi d'autres activités mathématiques)

Numérique et motivation

- Numérique plus « motivant » ?
- Feed-back neutres affectivement
- Anxiété mathématique réduite ?

L'expérimentation EXPIRE

Objectif de l'étude

Contribuer à la réflexion sur l'introduction de la pensée informatique comme véhicule pour l'apprentissage des mathématiques à l'Ecole primaire.

Evaluer les effets d'activités de programmation (une composante essentielle de la pensée informatique) pour l'acquisition de trois notions mathématiques : la division euclidienne, la décomposition additive et les fractions.

Séquences utilisant le logiciel *Scratch*

Scratch (logiciel développé au MIT)

- Développé pour les enfants
- Vrai langage de programmation
- Feed-back directs

- L'un des plus utilisés en classe

Langage de programmation visuelle

=> évite les erreurs de syntaxe

Exemple : travail sur la décomposition additive

Scratch 2 Offline Editor

Sequence 2

"total" avance lentement

total: 5047

Résultat

Scripts

Exécute

Algorithme

écriture algébrique :

$$5047 = 5 \times 1000 + 4 \times 10 + 7 \times 1$$

Exemple : travail sur la division euclidienne

The screenshot shows a Scratch workspace for an exercise titled "Exercice 4 : approcher 24 par sauts de 5". The workspace is annotated with several callouts:

- Programming area:** Points to the script area containing a "when green flag clicked" event, a "say" block, and "move" blocks.
- Button to execute the algorithm, the result of which is visible in the display area:** Points to a "when clicked" event block.
- Algorithm under development:** Points to the "move" blocks in the script area.
- Display area:** Points to a 10x10 grid of numbers. The number 24 is highlighted in the cell at row 8, column 5.
- Programming components provided for the current exercise:** Points to a collection of "move" blocks with various step sizes (1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12).

Consigne pour les élèves (Ex 1 à 3b) :

Ecrivez un programme dans Scratch permettant au lutin de se rendre exactement à la case cible.

Vous n'avez le droit d'utiliser qu'un seul type de saut à choisir entre 1 et 9.

Vous pouvez l'utiliser autant de fois que vous le souhaitez.

Attention : dans certains exercices il faut éviter des bombes !

Travailler les expressions « nombre de sauts », « il manque... » pour l'exprimer à l'aide de : $Cible = (Type\ de\ saut * Nombre\ de\ sauts) + Reste$.

Institutionnalisation : Introduction des termes de la division et écriture $D = (d*q) + r$

Parallèlement, exemple d'une situation de division euclidienne **pour le groupe contrôle**

Il faut découper des morceaux de 6 cm. Pour cela, nous disposons de 3 grands rubans de différentes longueurs : la 1^{ère} mesure 75 cm, la 2^e mesure 123 cm et la 3^e mesure 150 cm. Combien de rubans de 6 cm peux-tu découper dans chaque ruban ? Quelle longueur de papier reste-t-il à la fin de chaque découpage ?

Matériel

- Situation écrite au tableau avec une représentation schématique d'un morceau de 6 cm et des 3 rubans.
- bandes de 75 cm
- bandes de 123 cm
- bandes de 150 cm

Utilisation de la bande mesurant 75 cm : cette étape permet l'appropriation de la situation. Le matériel est utilisé comme support de raisonnement et de résolution, mais aussi comme moyen d'invalidation de leur résultat et comme support de raisonnement et de résolution. L'utilisation des bandes de 123 et 150 cm correspond à des phases de réinvestissement.

Petite précision :

- cette recherche ne vise pas à évaluer le logiciel *Scratch*
- mais l'effet de séquences d'enseignement des mathématiques via la programmation (réalisation d'algorithmes)

Méthode



Participants

46 écoles

Grenoble et agglomération
30 % Education Prioritaire

2576 élèves ←

50 % filles, 50 % garçons
47,5 % CM1
52,5 % CM2
Age moyen= 9,5 ans

109 enseignants (volontaires)

Age moyen = 44 (23 à 62)

Expérience moyenne = 17,5 (1 à 35)

89 % femmes

29 % formation scientifique

Pas très loin des 117 et
2808 visés pour un
MDE fixé a priori à 0,20

Répartition aléatoire des enseignants entre les deux groupes

- Apprentissage numérique (N = 1520)
- Apprentissage condition contrôle (N = 953)

Les enseignants **des deux groupes** ont reçu une formation de 3 heures dispensée par des membres de l'équipe de recherche en didactique des mathématiques.

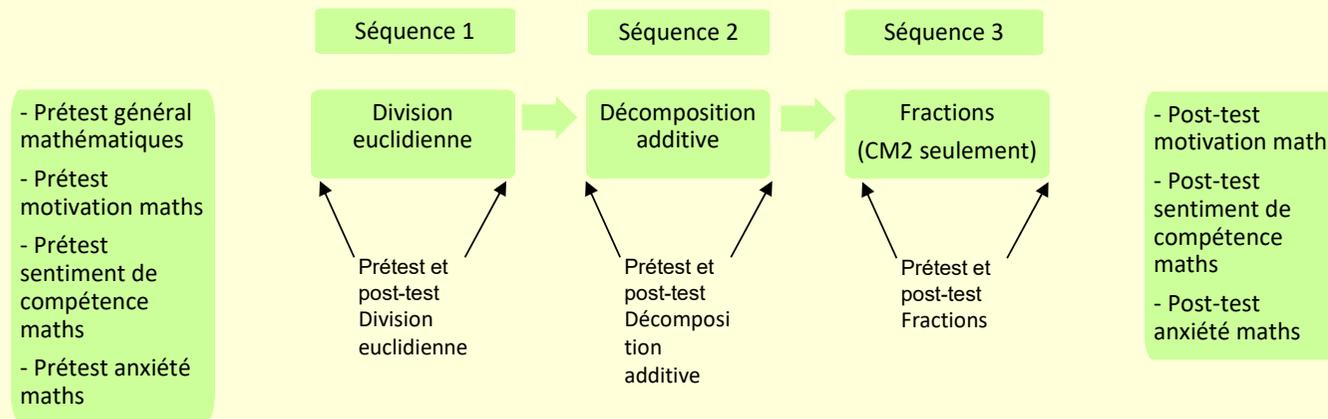
Cette formation consistait en une présentation des concepts mathématiques et des séquences pédagogiques.

Les enseignants **du groupe « programmation »** ont reçu une formation supplémentaire de 6 heures pour se familiariser avec *Scratch*.

Les enseignants ont reçu des liens leur permettant de télécharger la documentation spécifique à leur groupe, séquence par séquence (pour chaque séquence, environ 10 pages de détails pédagogiques pour l'enseignant et 6 pages d'exercices pour les élèves).

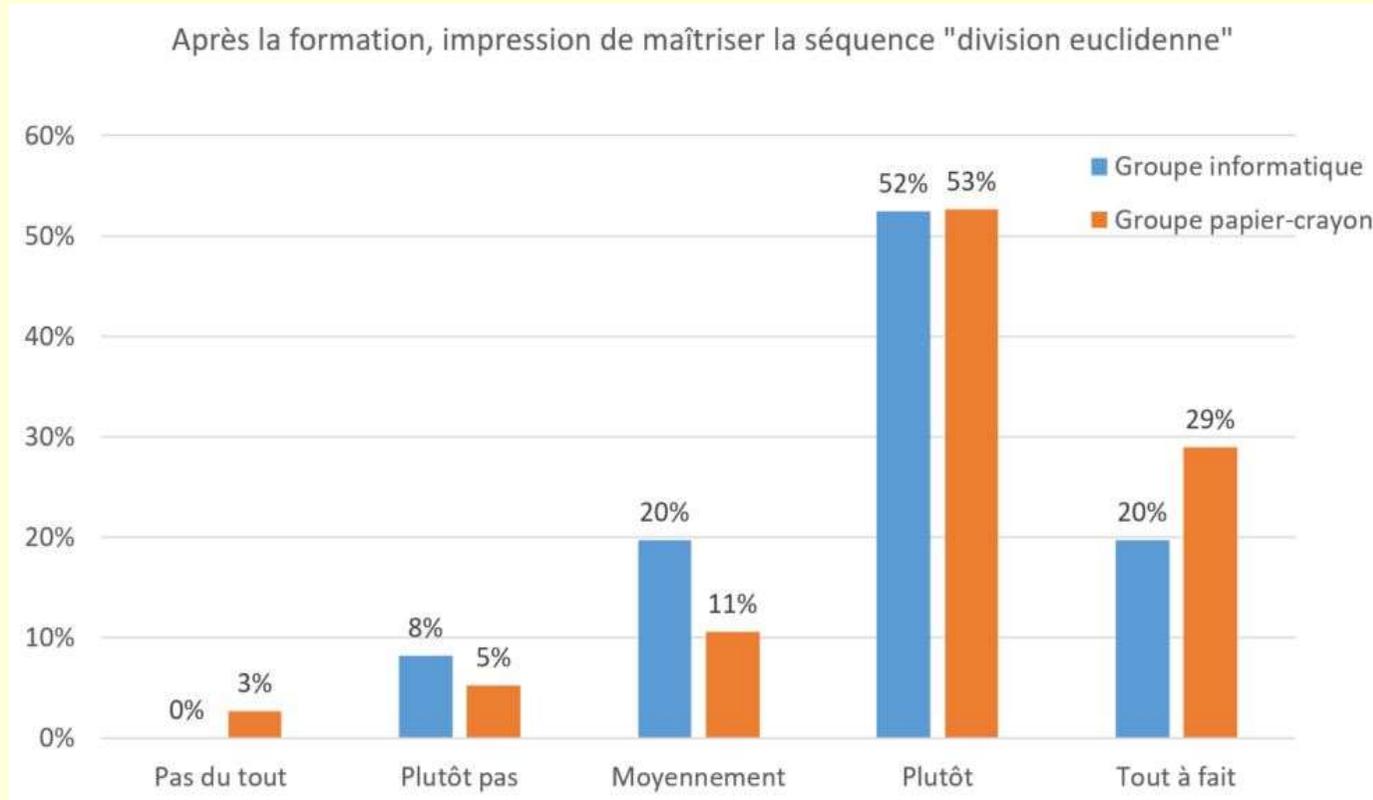
Le groupe contrôle a bénéficié de la formation *Scratch* et du matériel le semestre suivant.

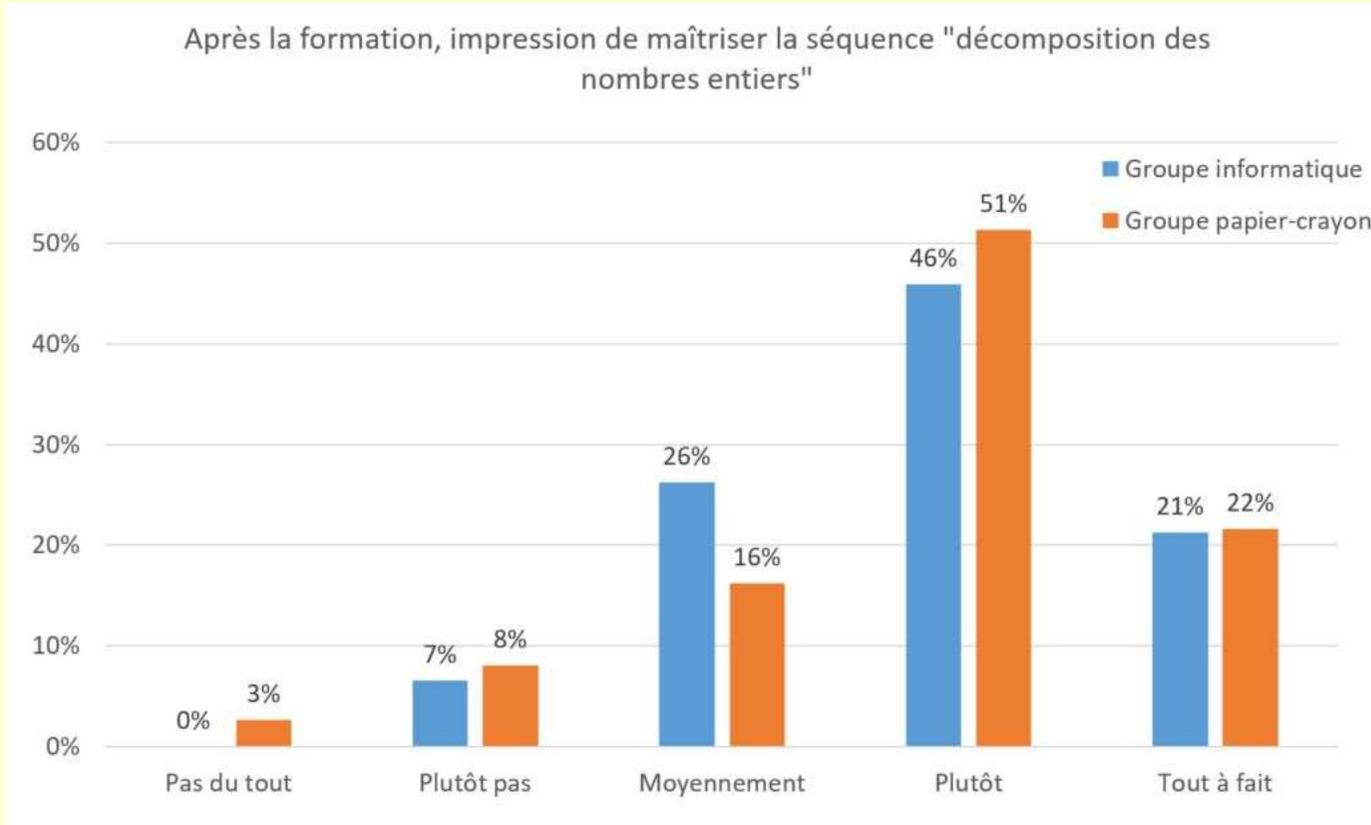
Chaque séquence est composée de trois séances d'une heure

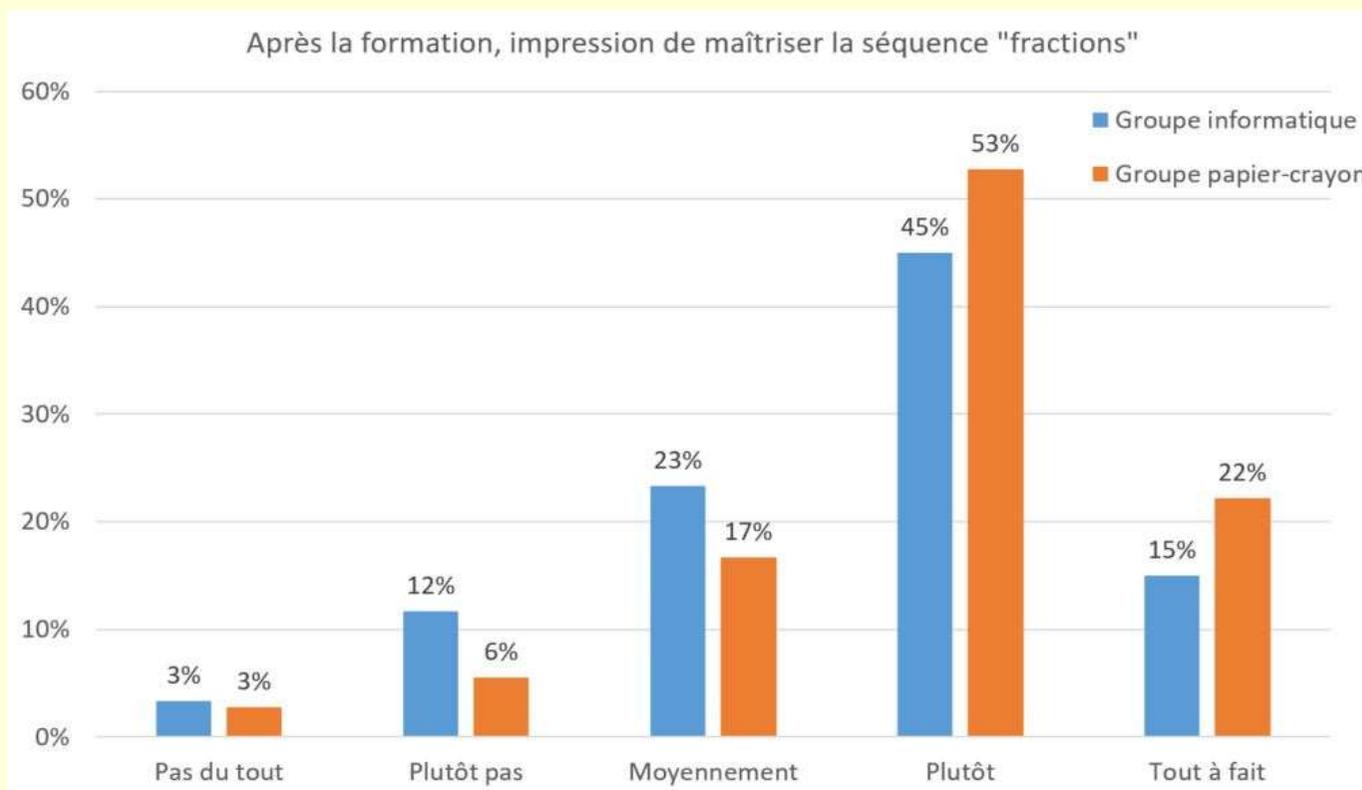


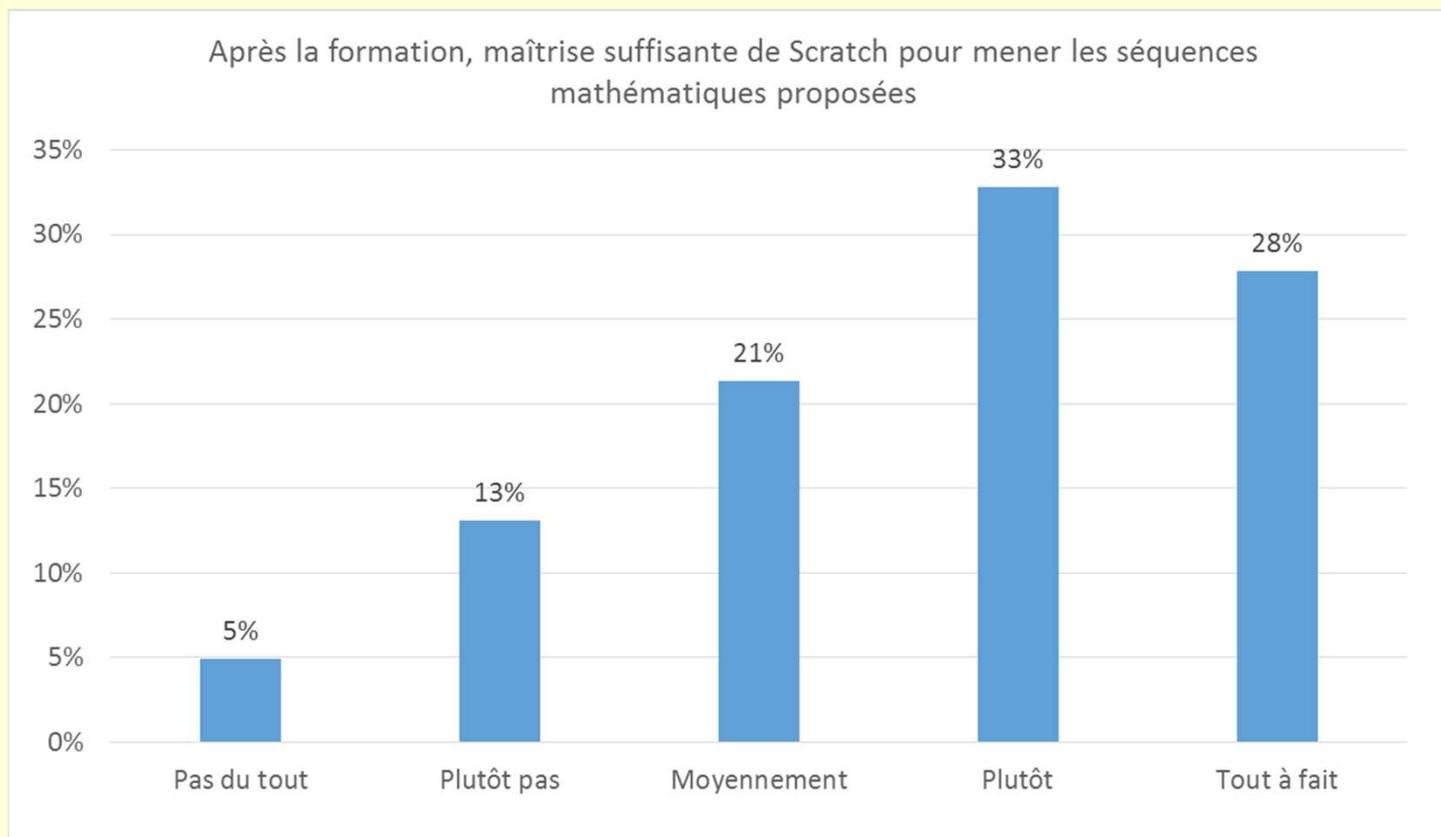
Expérimentation d'octobre 2017 à février 2018

Avis des enseignants sur la formation reçue









Résultats

Effets expérimentaux sur les acquis des élèves

	Effet expérimental
SEQ1 (division euclidienne)	-.16 (.06) $p < .01$

Effets expérimentaux sur les acquis des élèves

	Effet expérimental
SEQ1 (division euclidienne)	-.16 (.06) $p < .01$
SEQ2 (décomposition additive)	-.19 (.08) $p < .05$

Effets expérimentaux sur les acquis des élèves

	Effet expérimental
SEQ1 (division euclidienne)	-.16 (.06) $p < .01$
SEQ2 (décomposition additive)	-.19 (.08) $p < .05$
SEQ3 (fractions)	-.21 (.09) $p < .05$

=> Les élèves du groupe expérimental ont des acquis moins élevés que ceux du groupe contrôle en fin de séquence

- Pas d'effet d'interaction (notamment avec le niveau initial des élèves)
 - => le dispositif tend à « pénaliser » tout le monde
 - => il ne réduit pas les inégalités
-
- Pas d'interaction non plus avec le sentiment de maîtrise des enseignants au sortir de la formation EXPIRE

- Les effets sur la motivation, le sentiment de compétence et l'anxiété en maths sont non significatifs
- (mais peut-être aurait-il fallu les mesurer juste en fin de séance)

Discussion

- Le transfert d'apprentissage ne s'est pas produit comme prévu
- Résultats qui ne semblent pas liés à la maîtrise de *Scratch* car l'effet s'amplifie quelque peu dans le temps

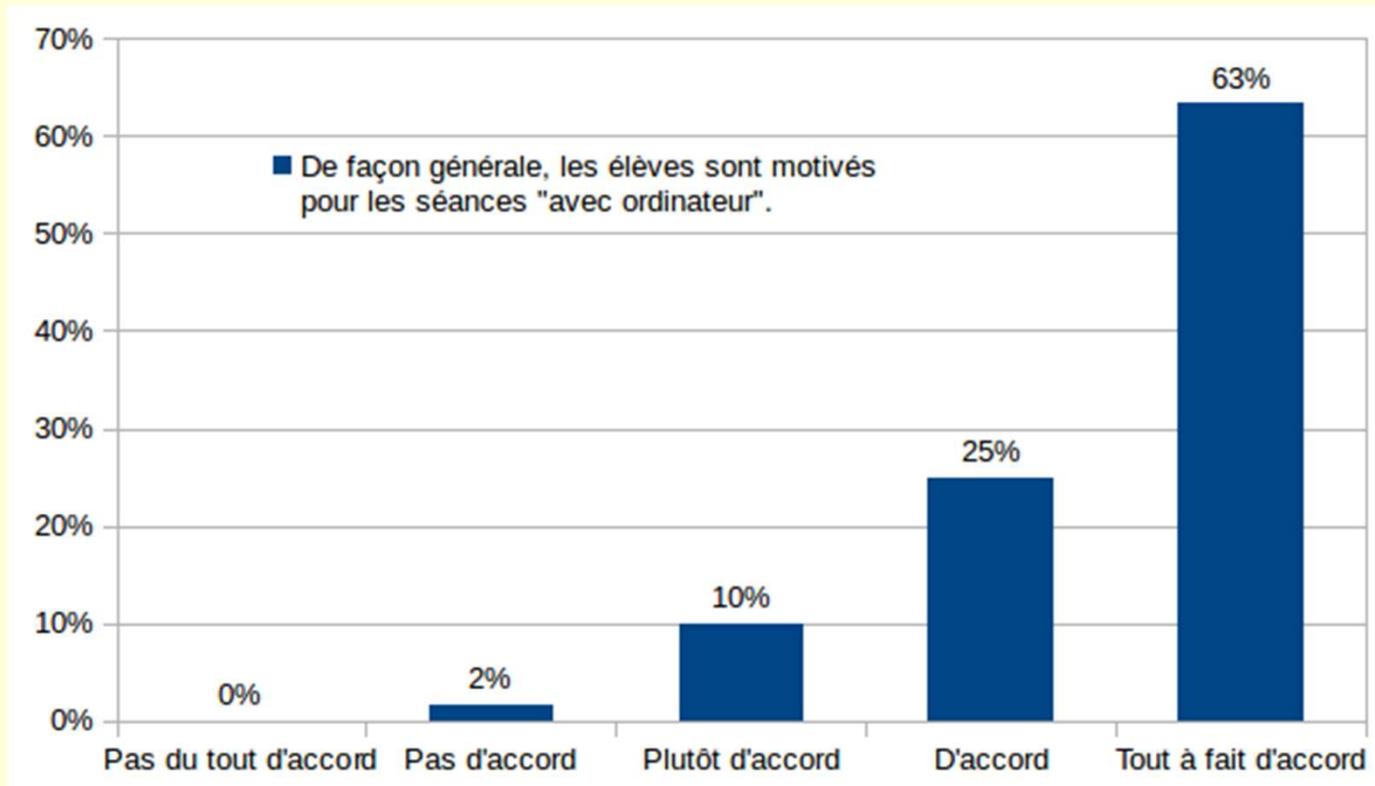
- Problème de transfert de la formulation algorithmique à la formulation algébrique ?
- Problème de transfert de la situation d'apprentissage à la situation d'évaluation ?

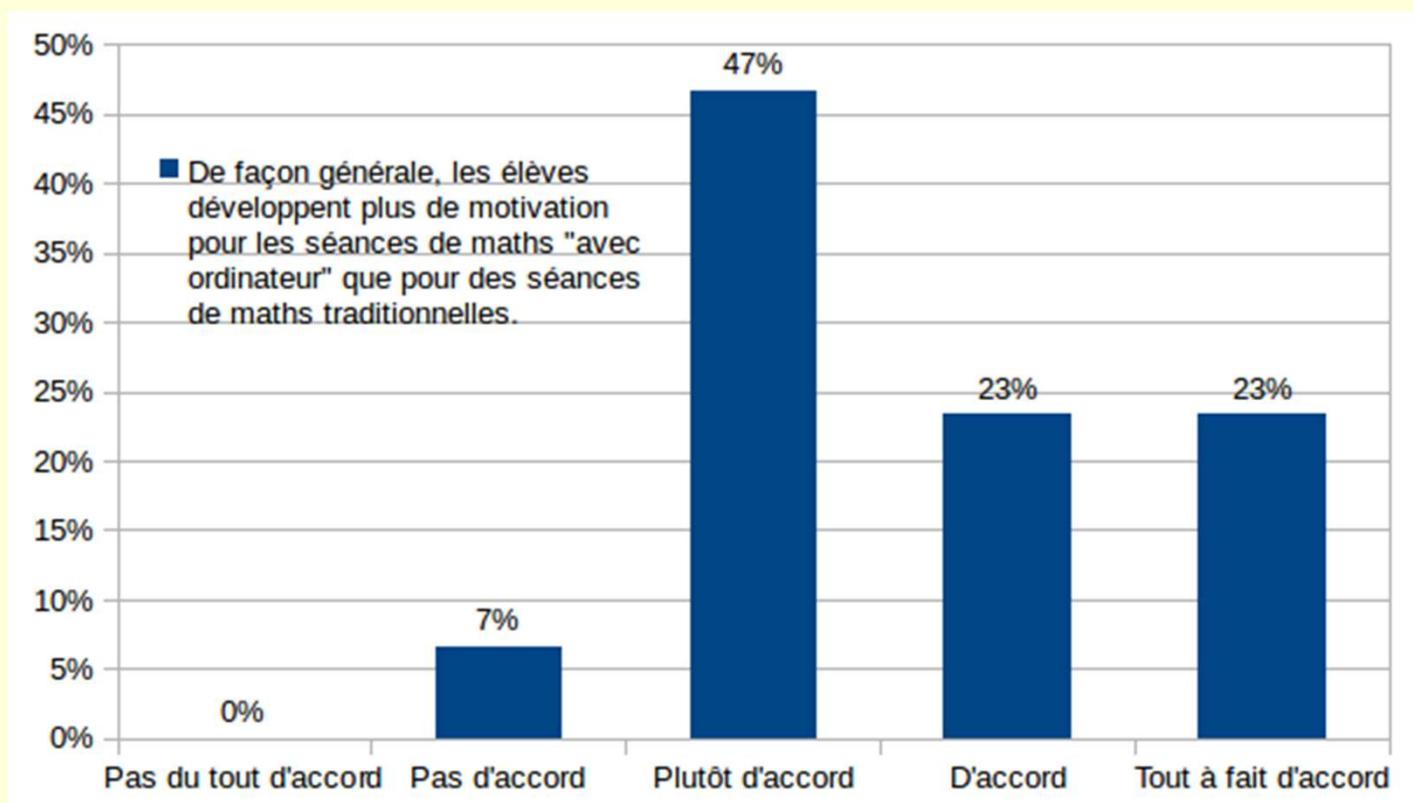
- Problème de charge cognitive (avoir à gérer deux registres, algorithmique et algébrique) ?
- Changement d'attention (palette d'options sur l'interface *Scratch* qui peut détourner les élèves de la tâche) ?
- Davantage de feed-back des enseignants sur l'utilisation du logiciel (et moins sur le contenu) ?
- Autre... ?

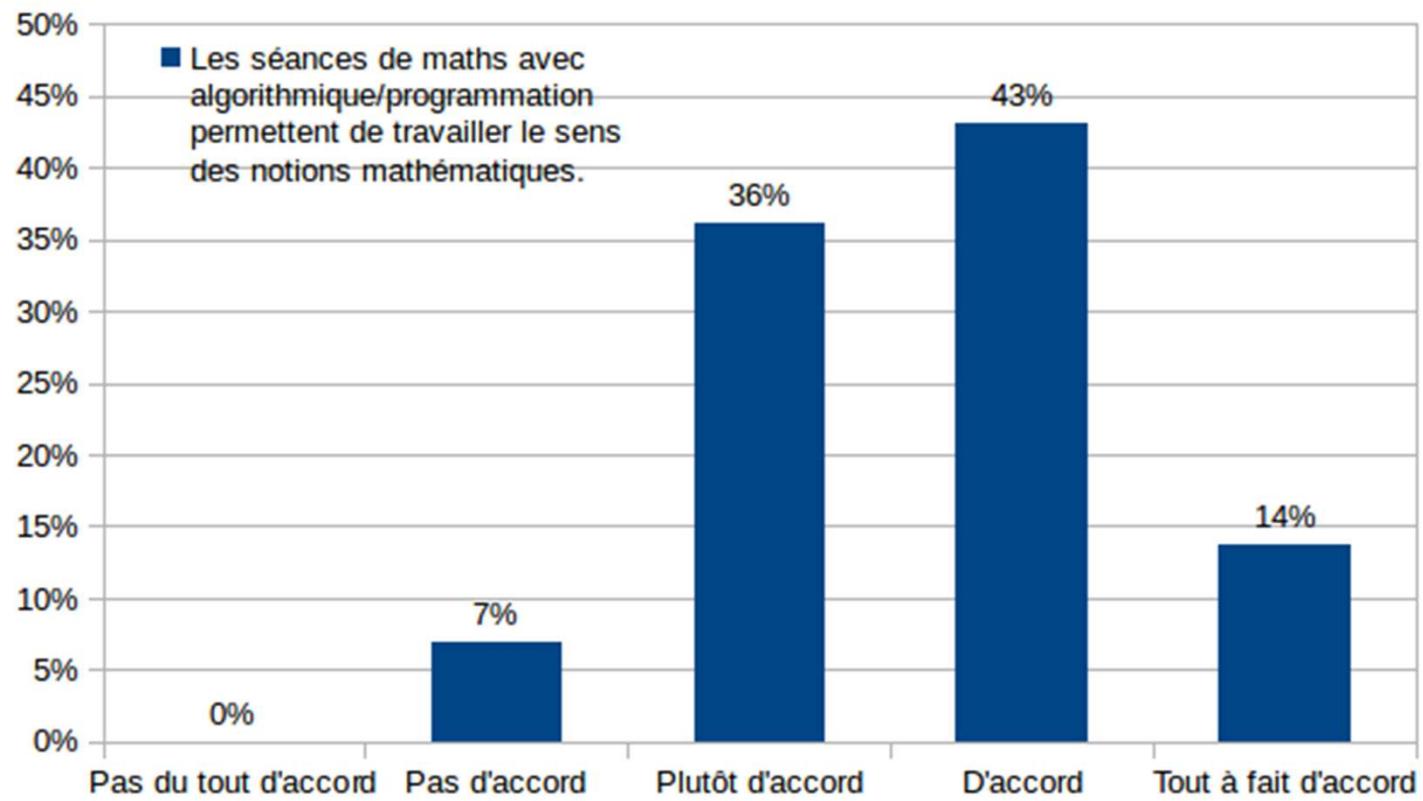
Implications pratiques

- Sans doute pas une bonne idée de **remplacer** (entièrement ?) les séances traditionnelles par des séances *Scratch*
- Mais peut-être utile pour des activités complémentaires (exercices supplémentaires, substitut du coin lecture, etc.)

Demander l'avis des enseignants aurait-il suffi ?







Une suite : EXPIRE II





**Merci pour votre
attention**

pascal.bressoux@univ-grenoble-alpes.fr