

# BILAN DU PROJET SPÉCIAL N°5

## MATHÉMATIQUES ET INFORMATIQUE

Responsables

MODESTE\* Simon – BUTEAU\*\* Chantal

Correspondante CS

CARON\*\*\* France

### I. INTRODUCTION

Se distinguant de la question de l'intégration des TICE dans l'enseignement des mathématiques, un projet spécial avait cours pour la première fois à l'EMF autour de la relation particulière qu'entretiennent les mathématiques et l'informatique, comme disciplines à part entière. Leurs liens sont différents de ceux que peuvent entretenir les mathématiques avec d'autres disciplines. En effet, en informatique comme en mathématiques, le formalisme symbolique occupe une place importante, les fondements s'appuient sur la logique, et une partie de la validation relève d'une activité de preuve. Ces points communs sont attestés par l'histoire des deux disciplines. Cependant, chacune de ces deux sciences a ses thèmes propres, et si l'informatique apporte des points de vue nouveaux en mathématiques ainsi que de nouveaux problèmes, il en résulte que plus en plus de connaissances mathématiques sont nécessaires en informatique.

Par ailleurs, l'informatique participe à l'évolution des rapports dans la recherche entre les mathématiques et les autres disciplines, par les nouveaux modèles et les simulations qu'elle permet d'envisager pour résoudre des problèmes complexes. Comme l'avance la Société mathématique européenne : “Together with theory and experimentation, a third pillar of scientific inquiry of complex systems has emerged in the form of a combination of modelling, simulation, optimization and visualisation.” (2011, p.2).

En mettant l'interdisciplinarité au cœur du débat ainsi que les enjeux sociétaux associés (compréhension du monde et de son évolution), le thème du colloque EMF 2018, *Mathématiques en Scène, des ponts entre les disciplines*, était donc particulièrement propice à explorer la relation qu'entretiennent mathématiques et informatique, entre elles et avec les autres disciplines, pour questionner leur enseignement et leur apprentissage.

Une telle exploration s'avère d'autant plus nécessaire que plusieurs réformes récentes ont mené à l'intégration de l'informatique dans les programmes scolaires obligatoires sous formes diverses (comme discipline scolaire indépendante, attachée aux mathématiques ou à d'autres disciplines, par des enseignants bivalents, etc.).

Sans s'y limiter, le groupe de travail souhaitait questionner, sur les plans épistémologique et didactique, les objets suivants :

- Programmation en mathématiques et mathématiques dans la programmation
- Pensée(s) informatique(s) et pensée(s) mathématique(s)

---

\* Université de Montpellier – France – [simon.modeste@umontpellier.fr](mailto:simon.modeste@umontpellier.fr)

\*\* Brock University – Canada – [cbuteau@brocku.ca](mailto:cbuteau@brocku.ca)

\*\*\* Université de Montréal – Canada – [france.caron@umontreal.ca](mailto:france.caron@umontreal.ca)

- Champs, concepts et objets partagés entre informatique et mathématiques, points communs et divergences
- Relation des mathématiques et de l'informatique aux autres disciplines, modélisation et simulation

Les questions abordées pouvaient traiter des choix de curriculums, des organisations d'enseignement (et de leurs réformes), des activités proposées en classe, de la conception de ressources, des apprentissages et difficultés des élèves, et conduire à questionner la formation initiale et continue des enseignants, à la fois sur les plans disciplinaire, épistémologique et didactique.

Sept contributions représentant 2 pays (France et Canada) ont répondu à l'appel. Témoignant de l'intérêt pour ce thème, plus de 60 personnes du colloque ont participé aux échanges du groupe.

Les contributions reçues ont conduit à organiser les travaux du groupe selon les axes suivants, lesquels constituent respectivement les thèmes de discussion des trois premières séances: i) Natures des pensées algorithmique/informatique, rapport aux mathématiques ; ii) Programmation pour les mathématiques, programmation par les mathématiques ; et iii) Objets, concepts, raisonnements à l'interface maths-info. Nous traiterons brièvement de chacun de ces axes dans les sections qui suivent. Nous poursuivrons avec des perspectives pour un prochain groupe de travail en guise de conclusion. Pour ce qui est de la quatrième et dernière séance, les discussions se sont poursuivies, en plus d'une présentation brève par des participants de la place de la programmation dans les programmes scolaires de différents pays– (voir l'Annexe 2).

## II. NATURE DES PENSEES ALGORITHMIQUE OU INFORMATIQUE, RAPPORT AUX MATHEMATIQUES

Avec l'intérêt suscité par l'article de Wing (2006), la nature de la pensée informatique (« computational thinking » ou « algorithmic thinking ») et ses liens avec la pensée mathématique, la pensée d'ingénierie et la pensée scientifique (Wing, 2008) nécessitent d'être éclairés.

Dans cette visée, Chantal Buteau a présenté un cadre conceptuel pour étudier la nature ainsi que le rôle et la place accordés à la pensée informatique dans l'enseignement des mathématiques au niveau postsecondaire. L'approche proposée prend en compte la discussion sur la pensée informatique en sciences et mathématiques, et elle s'appuie sur l'approche instrumentale pour décrire comment les étudiants en mathématiques de niveau universitaire peuvent s'approprier la programmation et s'engager dans un travail de pensée informatique pour les mathématiques, comme le feraient des mathématiciens. Certains participants ont souligné que, comme l'a révélé l'étude de Broley (2015), les pratiques informatiques ne sont pas entièrement acceptées par la communauté mathématique; si elles le sont davantage en mathématiques appliquées, la partie informatique du travail n'est pas systématiquement publiée, rendant invisible cet aspect du travail du mathématicien.

Marina Rafalska et Simon Modeste se sont intéressés aux conceptions d'élèves et d'étudiants à la transition secondaire-supérieur en France autour de l'algorithme et de l'algorithmique. À l'aide d'un questionnaire et d'une grille d'analyse basée sur un modèle épistémologique, ils montrent comment ces conceptions évoluent dans le temps et comment la dimension objet du concept se développe lors des premières années de l'enseignement supérieur. Cela confirme l'approche essentiellement basée sur l'algorithme en tant qu'outil au secondaire et la construction de la dimension objet du concept dans les premières années

d'université. L'analyse des réponses des étudiants montre aussi des conceptions variables selon les filières et options d'origine en particulier entre les cours de mathématiques, d'informatique et de sciences de l'ingénieur. Les auteurs mettent en avant que les conceptions des élèves/étudiants à la transition secondaire-supérieur permet aussi de confronter les curriculums et enseignements avec les conceptions effectivement construites.

Pour lancer la discussion collective suivant les deux communications, nous sommes partis de la définition de la pensée informatique (« computational thinking ») proposée par Wing (2008): « La pensée informatique est une sorte de pensée analytique. Elle partage avec la pensée mathématique les manières générales dont nous pourrions aborder la résolution d'un problème. Elle partage avec la pensée d'ingénierie les moyens généraux permettant de concevoir et d'évaluer un système vaste et complexe fonctionnant dans le respect des contraintes du monde réel. Elle partage avec la pensée scientifique les manières générales dont nous pourrions aborder la compréhension de la calculabilité, de l'intelligence, de l'esprit et du comportement humain. » (p.3717, traduit par nous).

Des participants ont fait ressortir la difficulté et même le risque d'établir des frontières entre pensée humaine, pensée mathématique, pensée algorithmique, et pensée informatique. Par exemple, si l'on réduit la pensée algorithmique à la capacité à traduire la résolution d'un problème en une suite d'instructions exécutable par n'importe quel opérateur, on court le risque d'associer une telle pensée à une pensée automatique, froide et stupide, et de l'opposer à la pensée mathématique, qui elle serait « noble », analogique, intelligente, intuitive. On retrouve cette idée chez Maurer (1998) : « In any event, algorithmics does not mean performing algorithms over and over by hand. Algorithms will be carried out more and more by machines or by person-machine combinations. Algorithmics is thinking about algorithms, not thinking like algorithms » (p.24).

### III. PROGRAMMATION POUR LES MATHÉMATIQUES, PROGRAMMATION PAR LES MATHÉMATIQUES

L'enseignement de la programmation (que ce soit avec des langages « classiques » ou des outils dédiés) peut aussi bien mettre en jeu des apprentissages mathématiques (programmation par les maths) que nourrir la dimension expérimentale des mathématiques (programmation pour les maths).

La contribution de Chaachoua, Tchnounike et Crisci présente un premier exemple de programmation pour les mathématiques, où les tâches de conception d'algorithme et de programmation ont pour intention d'aider à construire le sens de la division euclidienne ; sans qu'il y ait de concept informatique explicitement en jeu, le travail de l'algorithme est vu comme contribuant au développement de la pensée informatique. Si pour les auteurs, un environnement comme Scratch permet de traiter conjointement (voire même de faire coïncider) les phases d'écriture de l'algorithme et de programmation (car les élèves écrivent leur algorithme en agençant les blocs Scratch), ils font toutefois ressortir l'importance de confronter le programme à son exécution pour atteindre la notion mathématique en jeu, ce que ne permet pas la simple écriture d'un algorithme.

Un second exemple de programmation pour les mathématiques a été partagé par Jannick Trunkenwald, avec l'analyse a priori de la tâche de conception d'un programme de simulation d'une expérience aléatoire, à l'aide d'un générateur de nombres pseudo-aléatoires. À travers les deux niveaux de programmation impliqués (pour générer les résultats d'un échantillon et les différents échantillons), une telle tâche paraît contribuer au développement de la pensée informatique. Mais l'intention didactique est d'abord ici de faire apprécier la fluctuation

d'échantillonnage et d'ouvrir vers les intervalles de confiance. L'analyse de la tâche demande de considérer les techniques et technologies envisageables dans trois domaines distincts : probabilités, statistique et algorithmique.

Lors des échanges de cette séance, les questions suivantes ont été soulevées :

- Quelle institutionnalisation des concepts issus de l'informatique observe-t-on (ou devrait-on avoir) en école primaire ? Est-il possible de détacher l'activité de programmation de ces concepts ?
- Est-ce que le traitement au sein de l'informatique peut faire apparaître de nouveaux aspects des concepts mathématiques ?
- Quel rôle jouent les contraintes institutionnelles (notamment les programmes) dans la définition de situations d'apprentissage de l'algorithmique ?
- Est-ce que l'outil informatique aide toujours au développement de la connaissance mathématique ou est-ce que parfois il peut devenir un obstacle ?
- Au sujet de la représentation d'un algorithme dans un langage, le passage par l'écriture d'un algorithme semble indispensable pour l'écriture d'un programme ; l'est-il vraiment avec des langages comme Scratch ?
- Les apports de la didactique de l'informatique (Baron & Bruillard, 2001) ou de la psychologie de la programmation (Hoc et al., 1990) sur ces questions gagneraient à être pris en compte.

#### IV. OBJETS, CONCEPTS, RAISONNEMENTS A L'INTERFACE MATHÉMATIQUES-INFORMATIQUE

Quels que soient les contextes, la question de la relation aux mathématiques se pose, tant certains concepts ou outils sont proches.

Il convient donc d'aborder les questions liées aux interactions et aux interfaces entre mathématiques et informatique sans réduire l'une à l'autre, mais en pensant leurs objets et leurs activités comme un continuum polarisé par les deux disciplines. Cela passe par l'identification des concepts, objets, pensées, activités en jeu, et par le questionnement de leur enseignement et de leur apprentissage, en essayant de dépasser la juxtaposition des deux disciplines pour entrer dans une interdisciplinarité réelle. En particulier, les champs et concepts spécifiques qui se développent à l'interface des mathématiques et de l'informatique (mathématiques discrètes, algorithmique, ...) doivent être examinés ; cela peut se faire en adoptant tantôt les points de vue que porte l'informatique sur des concepts mathématiques classiques (variable, fonction, ...), tantôt les points de vue mathématiques sur les concepts de l'informatique.

Antoine Meyer et Simon Modeste ont mis en évidence le fait que la dichotomie et jeu de la devinette sont des thèmes présents dans les programmes, souvent vus comme allant ensemble. Ils distinguent derrière le terme français *dichotomie* les notions de recherche binaire (en algorithmique) et de *méthode de bisection* (d'après l'anglais). La dichotomie (bisection) peut être vue comme un moyen permettant d'apprendre, de mieux comprendre ou simplement d'utiliser le Théorème des valeurs intermédiaires, en offrant une construction possible pour traduire et compléter un théorème qui n'est pas constructif. Le choix d'un critère d'arrêt permet de comparer l'efficacité de différentes options (nombre maximal d'itérations,  $|a-b|$ ,  $|f'(c)|$ , ...). La recherche binaire fait appel au logarithme en base 2 et peut permettre d'aborder arbre de décision et écriture binaire des nombres. On peut aussi démontrer que l'algorithme de recherche binaire fonctionne toujours et est optimal. Et c'est un enjeu de l'informatique de prouver de telles choses.

Au-delà des mathématiques pour la démonstration en informatique et de l'informatique pour l'expérimentation en mathématiques, c'est bien une dialectique plus riche qu'il convient d'envisager entre ces disciplines. Les mathématiques permettent notamment d'ouvrir les boîtes noires présentes dans les outils de calcul et de représentation (ex. grapheurs munis de solveurs).

Dès lors qu'on ouvre aux approches algorithmiques, il faut pouvoir envisager leur coexistence (voire même leur concurrence) avec des raisonnements mathématiques plus traditionnels, de même que la possibilité d'émergence d'approches hybrides. L'analyse a priori gagne alors à anticiper les différentes techniques envisageables, et ménager un espace de négociation entre elles. Sylvain Beauvoir propose de s'appuyer sur une analyse des structures de contrôle (Balacheff et Margolinas, 2005), autant pour anticiper et analyser les raisonnements d'élèves que pour conduire l'institutionnalisation des savoirs.

Un travail des algorithmes et de la programmation en classe de mathématiques permet aussi de mieux comprendre par les mathématiques des applications importantes de l'informatique dans la vie quotidienne. Cela a pu être illustré avec l'exemple de la cryptologie, présenté par Emmanuel Volte (texte non présent dans les actes). En s'appuyant sur des programmes scratch, des élèves de niveau collège ont pu se familiariser avec la représentation étoilée de la fréquence des lettres pour identifier un cryptage de César (par simple décalage), enchaîner avec un encodage par méthode affine et aborder ensuite un encodage avec clé publique (méthode du sac à dos).

Les échanges de cette séance auront permis de dégager quelques concepts qui se situent à l'intersection (ou à l'interface) des mathématiques et de l'informatique ou qui, venant d'une discipline, bénéficient de l'éclairage de l'autre discipline. À l'intersection des deux disciplines, on réitère la présence des concepts de variable, de fonction, d'algorithme, etc. Du côté des concepts, idées ou même champs informatiques éclairés par les mathématiques, il y a d'abord évidemment l'algorithmique en général et certains de ses principes d'efficacité ("Diviser pour régner") qui conduisent à une analyse mathématique de la complexité, mais aussi plus particulièrement des champs tels que la théorie des graphes et autres domaines des mathématiques discrètes. Tout le champ de la géométrie computationnelle ou algorithmique pourrait constituer par ailleurs un domaine d'application des mathématiques, avec notamment le recours à la géométrie (repérée), aux transformations, à l'algèbre linéaire. Du côté des concepts ou idées mathématiques éclairés ou enrichis par l'informatique, on relève notamment les notions de division (euclidienne ou décimale) et le théorème des valeurs intermédiaires. On note toutefois une certaine domination du numérique dans les activités scolaires pensées et présentées dans ce groupe pour introduire à l'algorithmique.

## V. PERSPECTIVES

Au terme de nos discussions, nous retenons quelques perspectives qui demandent à être abordées dans un groupe de travail futur : approfondir l'articulation entre mathématiques et informatique (enjeux de logique et de preuve, relations faux amis / vrais parents de concepts et notions à l'intersection des deux disciplines, etc.) ; articuler avec les recherches en didactique de l'informatique ; examiner l'apport de l'informatique débranchée ; questionner la formation initiale et continue des enseignants ; suivre (et étudier) l'évolution des curricula (mathématiques et informatique) de niveaux scolaire et post-secondaire ; préciser les cadres théoriques (vocabulaires, objectifs, ...) ; envisager le rôle des mathématiques pour faire de l'informatique (programmation par les mathématiques).

On gagnerait à maintenir le caractère « spécial » de ce groupe transversal vis à vis des autres thématiques du colloque, afin de maximiser la participation à ce sujet ré-émergent.

## RÉFÉRENCES

- Balacheff, N. & Margolinas, C. (2005) cK $\phi$  Modèle de connaissances pour le calcul de situations didactiques. Dans : Balises pour la didactique des mathématiques, La Pensée Sauvage éditions, pp. 1 – 32.
- Baron, G.-L., & Bruillard, É. (2001). Une didactique de l'informatique ? Revue française de pédagogie(135), pp. 163–172. Consulté sur <http://www.jstor.org/stable/41201696>
- Brolley, L. (2015). La programmation informatique dans la recherche et la formation en mathématiques au niveau universitaire. Université de Montréal : Mémoire de maîtrise non publié, disponible sur : <https://papyrus.bib.umontreal.ca/xmlui/handle/1866/12574>
- Hoc, J. M., Green, T. R. G., Samurcay, R., J. Gilmore D. J. (Eds) (1990) Psychology of programming. London: Academic Press.
- Maurer S. B. (1998) What is an algorithm? What is an answer? In Morrow L. J., Kenney M. J. (Eds.), The teaching and learning of Algorithm in school mathematics, NCTM 1998 Yearbook, (pp.21-31). Reston, VA: NCTM.
- Société mathématique européenne (2011). Position paper of the European Mathematical Society on the European commission's contributions to European research. Consulté sur [http://ec.europa.eu/research/horizon2020/pdf/contributions/post/european\\_organisations/european\\_mathematical\\_society.pdf](http://ec.europa.eu/research/horizon2020/pdf/contributions/post/european_organisations/european_mathematical_society.pdf)
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33-35.
- Wing, J. M. (2008). Computational thinking and thinking about computing. Philosophical transactions of the royal society of London A: mathematical, physical and engineering sciences, 366(1881), 3717-3725.

## ANNEXE 1

## LISTE DES TEXTES DU SPE5 PAR ORDRE ALPHABETIQUE DU PREMIER AUTEUR

BUTEAU, C., MULLER, E., SACRISTAN, A.I. & MGOMBELO, J.

*Pensée informatique en enseignement des mathématiques post-secondaires : un cadre conceptuel*

CHAACHOUA, H., TCHOUNIKINE, P. & CRISCI, R.

*L'algorithmique et la programmation pour la construction du sens de la division euclidienne*

MEYER, A. & MODESTE, S.

*Recherche binaire et méthode de dichotomie, comparaison et enjeux didactiques à l'interface mathématiques – informatique*

MODESTE, S. & RAFALSKA, M.

*Conception de la notion d'algorithme à la transition secondaire-supérieur en France*

TRUNKENWALD, J.

*Entre probabilités et statistiques : un jeu algorithmique pour simuler la fluctuation d'échantillonnage*

## AFFICHES

BEAUVOIR, S.

*Analyse des outils de contrôle lors de la résolution algorithmique ou mathématique d'un problème*

VOLTE, E.

*Cryptographie au collège. Quels apports possibles ? Exemples de pratiques (poster présenté sans texte dans les actes).*

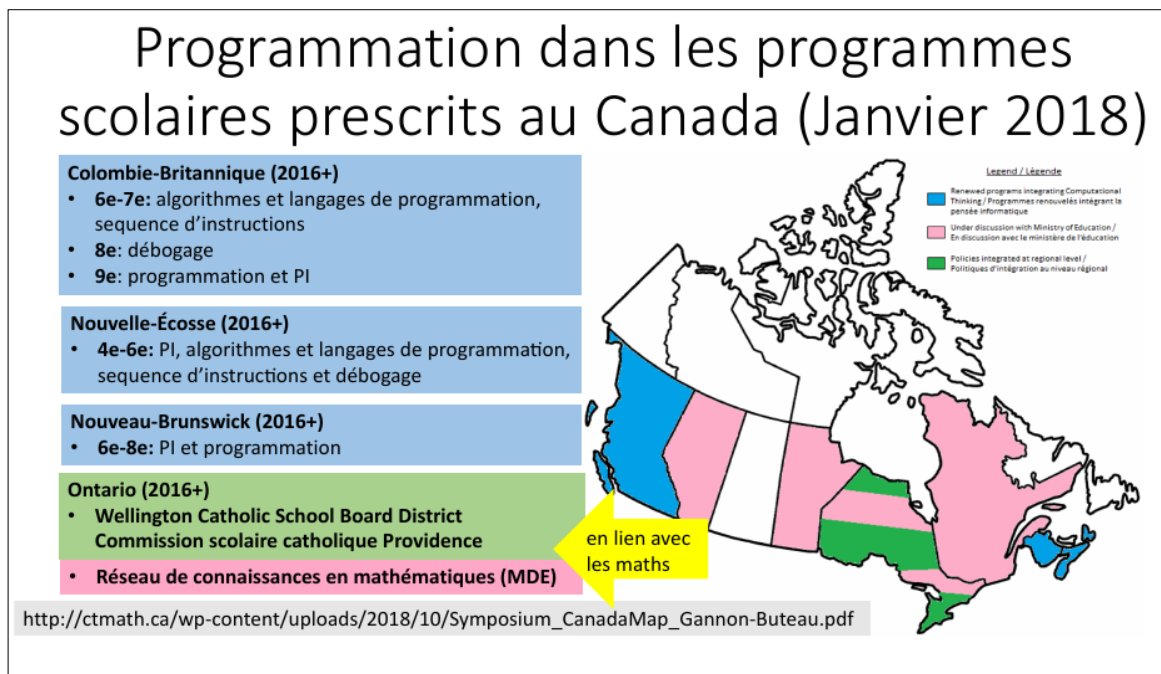
*Cette dernière contribution n'a pas donné lieu à un texte publié dans les actes*

## ANNEXE 2 – PLACE DE LA PROGRAMMATION DANS LES PROGRAMMES SCOLAIRES DE DIFFÉRENTS PAYS.

France :

- Au primaire (6 à 11 ans) Injonction. Un enseignant sur 2 du primaire n'a pas investi cette partie du programme. Problème de formation.
- Au collège (11 à 15 ans) enseignement partagé entre math et technologie.
  - scratch
  - variable, boucle, test
  - évalué au brevet APMEP
- En Lycée (15+) : la situation évolue beaucoup.
  - Actuellement : ICN en 2nde, ISN en Term S. Algorithmique en math. Aussi présent en SI ou en STI2D (carte arduino, ou RPI).
  - Demain : Multiplication par 10 des heures d'info (SIF). SNT (1 h 30) en 2nde. NSI (4 h en 1ère, 6 h en T). Algorithmique et programmation en Math.
- Une option au concours de recrutement d'enseignement de math

Canada (hors Québec) :





Québec :

