Titre: La pensée algorithmique comme une manière mathématique de penser

Auteurs: ST-CYR Marie-Frédérick, SQUALLI Hassane, VENANT Fabienne

Publication: Actes du huitième colloque de l'Espace Mathématique Francophone – EMF 2022

Directeur: Adolphe Cossi Adihou, Université de Sherbrooke (Canada/Bénin) avec l'appui des membres du comité scientifique et des responsables des groupes de travail et projets spéciaux

Éditeur: Les Éditions de l'Université de Sherbrooke

Année: 2023

Pages: 362 - 375

ISBN: 978-2-7622-0366-0

URI:

DOI:

La pensée algorithmique comme une manière mathématique de penser

ST-CYR[1] Marie-Frédérick – SQUALLI[2] Hassane – VENANT[3] Fabienne

Résumé – Bien que l'arrivée de l'ordinateur ait donné un nouveau souffle à l'activité algorithmique, celle-ci fait partie de l'activité mathématique depuis plusieurs centaines d'années. Dans ce texte, nous présentons la manière dont nous étudions la pensée algorithmique comme une forme de mathématique en nous appuyant sur la théorie de l'objectivation. Après avoir présenté brièvement notre caractérisation de la pensée algorithmique, nous expliciterons l'originalité de notre travail en le comparant avec des travaux précédents.

Mots-clefs : Didactique des mathématiques, pensée mathématique, pensée algorithmique, activité algorithmique, théorie de l'objectivation

Abstract – Although the advent of the computer has given new life to algorithmic activity, it has been part of mathematical activity for several hundred years. In this paper, we present the way we study algorithmic thinking as a form of mathematics based on the theory of objectification. After briefly presenting our characterization of algorithmic thinking, we will explain the originality of our work by comparing it with previous works.

Keywords: Didactic of mathematic, mathematic thinking, algorithmic thinking, algorithmic activity, theory of objectivation

1. Université de Sherbrooke, Canada, marie-frederick.st-cyr@usherbrooke.ca

2. Université de Sherbrooke, Canada, hassane.squalli@usherbrooke.ca

3. Université du Québec à Montréal, Canada, venant.fabienne@uqam.ca

Introduction

Le développement de la pensée mathématique et de ses différentes formes est au cœur de plusieurs programmes de formation dans le monde et au cœur de discussions en didactique des mathématiques. Alors que de nombreuses recherches s'intéressent à des formes de pensée associées à des domaines de contenus (par exemple la pensée algébrique est associée au domaine de l'algèbre), nous nous intéressons à une pensée qui est rattachée à un concept : celui d'algorithme. Ce concept, comme celui de fonction (Robert, 2018), est transversal puisqu'il est présent dans tous les domaines de contenus mathématiques.

La pensée algorithmique qui tantôt est appelée pensée informatique et tantôt pensée computationnelle fait actuellement l'objet de plusieurs recherches en didactique de l'informatique et en didactique des mathématiques en plus de s'être taillée une place dans plusieurs milieux scolaires à travers le monde. Dans ce texte, nous présentons un bref aperçu de l'évolution du concept d'algorithme en mathématique dans le temps en montrant notamment les changements qu'a apporté l'arrivée de l'ordinateur sur l'activité algorithmique en mathématique. Ensuite, en nous appuyant sur la théorie de l'objectivation (Radford) nous présentons la manière dont nous définissons la pensée algorithmique en mathématique ainsi que les grandes lignes de notre caractérisation de cette même forme de pensée mathématique. Nous terminons par discuter de l'originalité de notre travail en le comparant avec quelques recherches antérieures ayant porté un regard sur la pensée algorithmique dans un contexte mathématique.

L'activité algorithmique en mathématique au fil du temps

Bien avant la conceptualisation de l'algorithme, nous pouvons retrouver des traces laissant entrevoir la présence d'activités algorithmiques. En effet, on retrace dès le 9e siècle, dans les travaux du mathématicien Al-Khwarizmi, des algorithmes décrits dans le langage usuel. Certains des algorithmes décrits s'adressaient à des mathématiciens ou à des personnes ayant des connaissances mathématiques avancées en décrivant des procédures permettant de résoudre rapidement des équations. D'autres algorithmes étaient destinés aux citoyens qui n'avaient pas de connaissances mathématiques avancées dans le but de leur permettre d'effectuer des calculs mathématiques quotidiens sans se tromper (ou du moins en faisant le moins d'erreurs possible). En effet, Al-Khwarizmi[4] a repéré des problèmes de la vie courante, dont des problèmes liés aux échanges commerciaux, des problèmes d'arpentages et des problèmes de succession, puis il a réussi, avec ses compétences à reconnaitre des régularités et à généraliser, à proposer des solutions générales ceux-ci. Il a ensuite rédigé un manuel décrivant, de manière précise, les procédures à suivre pour effectuer des calculs de base de manière sûre et efficace. Un exemple d'un tel algorithme est l'algorithme d'addition.

4. Le mot algorithme provient d'une déformation latine du nom d'Al-Khwarizmi.

Par la suite, le concept d'algorithme s'est formalisé et a conservé une place importante dans l'activité mathématique. Nous pouvons, en outre, reconnaitre l'activité algorithmique par le fait qu'elle ait en son centre la présence d'une procédure systématique qui permette de répondre de manière juste à une famille de problèmes (Modeste, 2012 ; Ross, 2019). Jusqu'au milieu du 20e siècle, les algorithmes conçus étaient destinés à des humains leur permettant d'économiser du temps pour trouver la solution de certains problèmes. Puis, avec l'apparition de l'ordinateur, l'activité algorithmique a vu ses potentialités se multiplier. À partir de ce moment, les algorithmes n'étaient plus uniquement destinés à des humains, mais ils pouvaient être exécutés par une machine qui est plus efficace et qui fait moins d'erreurs que les humains. Or, ce changement a ouvert la porte à de nouvelles activités mathématiques telles que la preuve automatisée, l'expérimentation sur plusieurs exemples ou encore la vérification de conjectures (Broley, 2015). De plus, l'activité informatique a amené l'importance d'étudier la complexité et l'efficacité d'un programme, donc d'un algorithme. Cette activité demande la mobilisation de concepts mathématiques avancés et est devenue une activité mathématique importante en informatique puisqu'elle permet de comparer des algorithmes et, ultimement, de les optimiser.

Si au départ l'objectif de la conception d'algorithmes était de répondre à un besoin social, en permettant aux individus d'effectuer des calculs quotidiens, et au besoin des mathématiciens d'avoir des procédures efficaces pour effectuer certaines tâches ; il est clair que ce besoin a évolué. En effet, l'omniprésence d'outils pour calculer (calculatrices, ordinateurs, cellulaire, etc.) rend moins pertinente la capacité à effectuer très rapidement et sans faire d'erreurs des calculs. Toutefois, des individus qui ont développé une manière de penser leur permettant de créer les algorithmes pour que les machines opèrent sont très recherchés. En ce sens, le système d'éducation, qui vise à répondre aux besoins d'une société à un moment donné, doit préparer les élèves aux besoins actuels et futurs (Squalli, 2000). Par conséquent, les élèves doivent développer leur pensée algorithmique dans différentes sphères, dont en mathématique où les algorithmes ont émergé. Deux questions se posent alors : Quelle est cette forme de pensée spécifique en mathématique ? Quelles sont les caractéristiques de cette forme de pensée mathématique ? Comment aider les élèves à la développer ?

Pour entamer la discussion, nous allons présenter notre manière d'approcher la pensée par l'activité pour ensuite présenter les principales activités algorithmiques en mathématiques et terminer par distinguer le travail que nous effectuons des travaux précédents qui traitent de la pensée algorithmique.

Notre vision de la pensée à travers la théorie de l'objectivation

En nous intéressant à la pensée algorithmique en mathématique, nous nous inscrivons dans les recherches qui s'intéressent aux différentes formes de pensée en mathématiques. Nous approchons la pensée algorithmique dans le cadre de la théorie de l'objectivation (TO) (Radford, 2011), une théorie socioculturelle d'origine vygotskienne. Dans la TO le processus d'enseignement-apprentissage est considéré comme un travail conjoint, entre l'enseignant et les élèves, de rencontre avec les savoirs mathématiques chargés historiquement et culturellement. Afin de situer notre projet, nous précisons d'abord ce lien qu'unit la pensée et l'activité pour ensuite distinguer brièvement la pensée algorithmique dans son sens anthropologique et dans son sens subjectif en mettant l'accent sur la pensée anthropologique qui est au cœur de notre travail.

La pensée et l'activité algorithmique

Ce qui nous a permis de reconnaitre la pensée algorithmique dans le travail d'Al-Khwarizmi n'est pas la conceptualisation de l'algorithme ; ce sont les traces qu'il a laissées. Ces traces rendent compte d'une activité et d'un système sémiotique de significations culturelles qui « inclut des signifiés culturels tels que des conceptions autour des objets mathématiques [et algorithmiques] (leur nature, leur mode d'existence, leur relation avec le monde concret, etc.) et des modèles sociaux de production de signifiés » (Radford, 2011, p. 58). C'est donc dire que les algorithmes conçus par Al-Khwarizmi que nous pouvons retrouver aujourd'hui dans divers documents représentent beaucoup plus que la procédure permettant de calculer, par exemple, la somme de deux nombres; ils portent en eux une partie de la culture de l'époque et de l'activité, donc de la pensée, d'Al-Khwarizmi. En somme, la pensée se matérialise dans l'activité. La pensée et l'activité sont pour nous deux faces d'une même médaille.

La pensée au sens anthropologique et au sens subjectif

La pensée anthropologique relève d'une synthèse codifiée des pratiques sociales, culturelles et historiques (Radford, 2015). Elle s'est construite au fil du temps au travers de « l'agir des individus et [des] pratiques sociales à l'intérieur desquelles ces individus agissent » (Radford, 2015, p. 336). Cette synthèse résulte de la pratique sociale, les singularités des actions se voient *reflétées* dans ce qui devient reconnu comme une même manière d'agir et de réfléchir, se constituant ainsi en un prototype d'actions et de réflexions (Radford, 2015). La pensée est dès lors une capacité toujours latente d'agir ou de réfléchir d'une certaine manière. La pensée subjective, celle du sujet, est l'*actualisation*, la *concrétisation* ou la *matérialisation* de la pensée au sens anthropologique, d'une potentialité culturelle (Radford, 2015).

Pour l'instant, nous considérons uniquement la pensée anthropologique afin de décrire, le mieux possible, la pensée algorithmique en mathématique qui s'est construite au travers des cultures pendant plusieurs centaines d'années. Toutefois, nous ne pourrons présenter pleinement la pensée algorithmique dans son entièreté puisqu'elle ne peut être totalement saisie dans son tout. Nous la décomposerons donc pour la rendre accessible.

La pensée algorithmique et le concept d’algorithme

À la lumière de notre vision de la pensée, telle que la définit la TO, nous définissons *la pensée algorithmique en mathématique comme une manière d’agir et de réfléchir dans des activités mathématiques faisant intervenir au moins un algorithme*.

Un algorithme a trois caractéristiques essentielles : (1) une série d’instructions élémentaires opérationnelles pouvant inclure des actions répétées et des prises de décisions ou la sélection d’éléments menant à différentes suites d’actions (ces décisions et ces sélections sont basées sur des conditions) ; (2) doit se terminer en un nombre fini d’étapes (Modeste, 2012) et (3) lorsqu’exécuté correctement, donne la bonne réponse (Bråting et Kilhamn, 2021).

De plus, pour nous, le fait que l’algorithme soit adressé à un humain ou à une machine n’a pas d’importance. Cet aspect pourra influer la manière de le représenter, mais il n’en changera pas l’activité. Par conséquent, à l’instar de plusieurs chercheurs (Denning, 2017 ; Knuth, 1985 ; Modeste, 2012), nous incluons l’automatisation à notre conceptualisation de l’algorithme et donc, de la pensée algorithmique en mathématique.

Les classes d’activités algorithmiques en mathématique

Notre définition de la pensée algorithmique en mathématique s’appuie sur des activités algorithmiques en mathématiques. Or, quelles sont ces activités ? En nous appuyant sur notre définition, nous avons identifié deux critères pour identifier des classes essentielles d’activités algorithmiques en mathématiques : (1) il s’agit d’une activité mathématique et (2) l’activité fait intervenir au moins un algorithme.

En nous appuyant sur ces critères, nous avons identifié trois classes d’activités algorithmiques en mathématiques essentielles : la mobilisation d’un algorithme existant, la conception d’un algorithme et l’étude d’un algorithme. Ces trois classes d’activités algorithmiques en mathématiques, bien que présentées de manière distincte, sont interreliées les unes aux autres. Il est possible qu’à l’intérieur d’une même activité une personne conçoive tantôt un algorithme et qu’ensuite elle l’étudie et qu’elle le mobilise. Ces classes ne sont pas perméables, mais nous permettent de distinguer trois activités algorithmiques essentielles que nous décrirons tour à tour.

La mobilisation d'un algorithme existant

La mobilisation d'un algorithme existant se réfère aux activités mathématiques où une personne fait appel à un algorithme qui existe déjà pour réaliser une tâche ou résoudre un problème mathématique. Elle peut le transposer dans un langage différent (par exemple le transposer du langage mathématique vers un langage de programmation) ou elle peut l'utiliser de manière intentionnelle pour exécuter une tâche précise. Cette idée d'intention est importante pour que l'activité soit considérée comme algorithmique. C'est avec cette intention que la personne a le potentiel de rencontrer la pensée algorithmique.

Si nous remontons à l'époque d'Euclide (qui précède celle d'Al-Khwarizmi), nous retrouvons un algorithme qu'il a proposé pour calculer le plus grand commun diviseur de deux nombres naturels *a* et *b* ($a > b$). Voici cet algorithme écrit dans un langage mathématique actuel : *Soit* r_i *un entier le reste de la division euclidienne de a par b et* x_i *un entier tel que* $a=bx_1+r_1$. *Si* $r_1=0$, *alors pgcd(a,b)=b, sinon pgcd(a,b)= pgcd(b, r1). Alors, on reprend jusqu'à ce que* $r_n=0$ *comme suit :*$b= r_1x_2+r_2$; $r_1= r_2x_3+r_3$ *; etc. Lorsque* $r_n=0$, *alors pgcd(a,b)=* $r_{(n-1)}$.

Une personne qui serait dans une activité de mobilisation d'un algorithme pourrait adapter cet algorithme dans un langage de programmation. Contrairement à une personne qui utiliserait machinalement l'algorithme sans réfléchir, en le transposant elle agirait de manière intentionnelle et entrerait alors dans une activité algorithmique. Par conséquent elle développerait sa pensée algorithmique.

La conception d'un algorithme

La conception d'un algorithme se réfère aux activités mathématiques dans lesquelles une d'un personne construit un algorithme en s'appuyant (ou non) sur un (ou de plusieurs) algorithme(s) existant(s). La conception d'un algorithme peut avoir diverses visées dont permettre de modéliser ou de tester une solution ou d'émettre une conjecture. Toutefois, peu importe l'objectif de l'activité, la personne qui conçoit l'algorithme est amenée à décomposer un problème et/ou une solution en étapes élémentaires, à identifier des régularités et à généraliser lorsque cela s'applique.

Dans l'exemple de l'algorithme d'Euclide, Euclide qui conçu cet algorithme, a identifié des opérations qui se répétaient ($b= r_1x_2+r_2$) jusqu'à obtenir un reste nul. Il a donc détaillé sa stratégie en décomposant les étapes de sorte qu'elles soient les plus simples possibles pour qu'elles soient facilement reproductibles et en généralisant sa solution à tous les calculs de pgcd possibles. Par ailleurs, une élève pourrait trouver le pgcd de deux nombres par recherche exhaustive pour arriver à identifier les deux grandes étapes : (1) lister les diviseurs de chacun des nombres et (2) déterminer le nombre commun le plus grand. Elle pourrait aussi choisir de réutiliser un algorithme informatisé qu'elle a déjà fait pour lister les diviseurs d'un nombre afin de le compléter et de l'adapter pour qu'il puisse déterminer le plus grand diviseur commun aux deux nombres. Cette élève est alors dans une activité de conception dans laquelle elle a mobilisé et adapté un algorithme qui existait déjà.

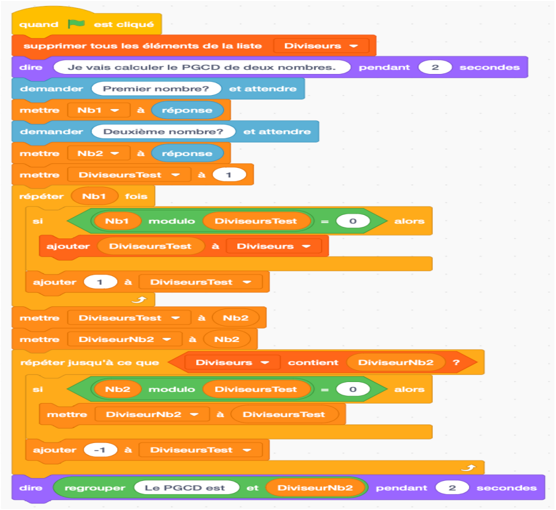

Figure 3 - Programme permettant de lister tous les diviseurs d'un nombre (encadré), puis adapté pour calculer le PGCD

L'étude d'un algorithme

L'étude d'un algorithme est l'activité mathématique dans laquelle une personne réfléchit à la structure de l'algorithme, à savoir notamment s'il répond bien au problème initial en un nombre fini d'étapes, tout en fournissant une réponse juste lorsqu'il est exécuté adéquatement. Cette activité peut aussi être une activité de comparaison des algorithmes entre eux en comparant leur structure, mais aussi en s'intéressant à leur complexité ou à leur efficacité[5]. La comparaison de l'algorithme de division tel qu'Euclide l'a proposé et l'algorithme par recherche exhaustive pourrait être une activité d'étude d'un algorithme. Ces deux algorithmes fonctionnent, mais il serait intéressant de comparer les processus et de faire ressortir leurs ressemblances et leurs différences. Nous pourrions aussi nous intéresser à savoir lequel est le plus efficace et dans quelle situation c'est bien le cas.

Ces trois classes d'activités, telles que présentées, peuvent apparaitre à des moments distincts, mais peuvent aussi apparaitre simultanément. En effet, dans une même activité, il est possible d'à la fois concevoir un algorithme et d'en faire l'étude pour essayer de le rendre plus efficace par exemple. Également, des sous-classes d'activités telles que la programmation et la généralisation peuvent apparaitre dans différentes classes d'activités. Or, pour distinguer les activités les unes des autres, nous allons nous appuyer sur trois dimensions.

5. Les notions de complexité et d'efficacité surviennent généralement lorsque l'algorithme est programmé. Or, l'activité nécessaire pour évaluer ces notions est une activité mathématique dans laquelle les propriétés des algorithmes sont étudiées.

Dimensions de la pensée algorithmique en mathématique

Bien que la définition de la pensée algorithmique en mathématique nous éclaire sur cette forme de pensée, elle n'est pas opératoire. Or, à l'instar de travaux sur la pensée algébrique (Squalli, 2015, 2020) et sur la pensée fonctionnelle (Robert, 2018), nous décrivons, de manière non exhaustive, la pensée algorithmique en mathématiques selon trois dimensions : les concepts en jeu et leur signification (opérateur, variable, symbole d'égalité, fonction), les raisonnements utilisés (séquentiel, itératif, récursif, conditionnel, analyse de l'erreur) et les modes de représentation et les manières d'opérer sur celles-ci (mathématique, informatique, pseudocode, passage d'un registre à l'autre). Nous décrivons globalement l'activité algorithmique en mathématique dans ce texte, mais nous pourrions également distinguer ces dimensions pour chacune des classes d'activités. Ce travail est prévu dans le cadre de notre projet de maitrise.

Originalité de notre travail

Le travail que nous avons entamé sur la pensée algorithmique en mathématique contribue aux réflexions sur la pensée algorithmique en mathématique et en informatique. Or, notre travail se distingue de ces travaux pour trois raisons principales : (1) nous considérons la pensée algorithmique comme une forme de pensée mathématique, (2) nous nous inscrivons dans une approche socioculturelle selon laquelle nous approchons la pensée par l'activité et (3) nous considérons trois dimensions pour décrire la pensée algorithmique dans son sens anthropologique. Nous présentons chacun de ces éléments qui rendent notre recherche originale en posant une réflexion critique sur certains travaux clés ainsi que sur notre propre projet.

La pensée algorithmique comme une forme de pensée mathématique

Nous regardons la pensée algorithmique comme une forme de pensée mathématique. Nous ne nions pas qu'elle existe dans d'autres sphères d'activités (notamment en sciences informatiques), mais nous nous intéressons spécifiquement à l'activité algorithmique qui a lieu dans des activités mathématiques. Par conséquent, notre travail ne vise pas à faire les contours de la pensée algorithmique en mathématique en la distinguant et en la liant à la pensée informatique comme l'ont fait Knuth (1985), Modeste (2012) et Shute (2017), mais simplement à décrire cette forme de pensée mathématique. Nous regardons alors la pensée algorithmique comme une forme de pensée mathématique. Cette posture est appuyée par l'histoire de l'activité algorithmique qui a débuté en mathématique et qui continue d'être au cœur de celle-ci.

Nous nous distinguons également de recherches qui visent à caractériser la pensée informatique ou computationnelle, pour ensuite identifier les caractéristiques qui sont aussi présentes en mathématiques. Un exemple d'une telle recherche a été menée par Kallia et son équipe (2021) qui ont fait une revue de la littérature pour identifier les caractéristiques de la pensée informatique, puis qui ont, à l'aide d'une étude de Delphes, identifié celles qui étaient aussi présentes dans l'activité mathématique. Un second exemple est le travail de Weintrop et de son équipe (2016) qui ont identifié des activités computationnelles pertinentes pour les cours de mathématiques et de sciences. Ils ont notamment identifié la résolution de problèmes computationnels comme centrale à leur taxonomie de la pensée computationnelle pour les classes de mathématique et de science. Ces travaux permettent de mettre en lumière plusieurs caractéristiques de la pensée algorithmique en mathématique qui nous seront fort utiles, mais ne décrivent pas l'activité algorithmique comme une manière mathématique de penser.

Notre manière d'approcher la pensée par l'activité

En inscrivant notre travail dans la théorie de l'objectivation, nous soutenons que la pensée algorithmique en mathématique est une manière d'agir et de réfléchir dans des activités algorithmiques en mathématique. La pensée algorithmique, dans son sens anthropologique, est construite, et continue de se construire, par l'ensemble des interactions qui se sont déployées dans des activités algorithmiques. Nous croyons donc que c'est dans l'activité algorithmique en mathématique que la pensée algorithmique en mathématique prend forme.

En adoptant cette position, nous nous distinguons du travail de Modeste (2012) qui a identifié des problèmes fondamentaux pour l'algorithmique. Ces problèmes sont pertinents et ont probablement le potentiel de générer une activité algorithmique, mais ce n'est que dans l'activité que ce potentiel peut, ou non, s'actualiser. Par conséquent, nous soutenons que ces problèmes fondamentaux pour l'algorithmique ne sont pas tributaires d'une activité algorithmique réelle. Ils ont probablement le potentiel de générer cette activité, mais à eux seuls, ils ne peuvent pas être ou décrire l'activité algorithmique elle-même.

De plus, notre posture nous amène à décrire la pensée algorithmique et à la découper de manière différente. En effet, en identifiant des activités algorithmiques, nous ne décomposons pas la pensée selon des tâches ou selon des domaines mathématiques auxquels ces tâches peuvent appartenir ; nous la découpons selon des classes d'activités algorithmiques en mathématiques, selon les manières d'agir et de réfléchir dans ces activités. Ce choix nous permet d'inclure, contrairement à Knuth (1974), la complexité et l'opération d'affection dans l'activité algorithmique lorsque celle-ci s'inscrit dans une activité mathématique.

Les trois dimensions de la pensée

En décrivant la pensée algorithmique en mathématique selon trois dimensions, nous distinguons clairement les concepts, les raisonnements utilisés et les modes de représentation. Ce choix ressemble à celui fait par Brennan et Resnick (2012) qui, dans leur caractérisation de la pensée computationnelle, ont identifié des concepts, des pratiques (qui se rapprochent des raisonnements dans notre cas) et des perspectives. Les deux premières dimensions rejoignent les nôtres, alors que la troisième, bien qu'elle inclue le langage, inclut aussi le questionnement qui n'est pas un mode de représentation.

Par ailleurs, notre choix de distinguer trois dimensions se distingue de recherches qui proposent des caractérisations sans tenir compte du type de caractéristiques sélectionnées. Le travail de Kallia et de son équipe (2021) illustre bien cette différence. En effet, cette équipe a élaboré une caractérisation de la pensée computationnelle pour l'enseignement des mathématiques selon trois dimensions : (1) La résolution de problèmes est le but fondamental de l'enseignement des mathématiques dans laquelle la pensée computationnelle est incluse. (2) Les processus de pensée incluant (mais ne se limitant pas à) l'abstraction, la décomposition, la capacité à reconnaitre des régularités, la pensée algorithmique, la modélisation, la logique, la pensée analytique, la généralisation, l'évaluation de solutions et de stratégies. (3) L'écriture de solutions à des problèmes mathématiques de telle sorte qu'ils puissent être transférés et traités (transposés) par une autre personne ou par une machine.

Nous constatons que le premier et le troisième point présentent, séparément, les éléments clés de la définition proposée par Wing (2010). Elle définit la pensée computationnelle comme « the thought processes involved in formulating problems and their solutions so that the solutions are represented in a form that can be effectively carried out by an information-processing agent » (Wing, 2010, p.1). Le second point présente des processus de pensée. Certains sont des types de pensée (pensée algorithmique, pensée analytique), alors que d'autres se réfèrent à des processus généraux en mathématique (abstraction, généralisation). Bien que non opératoire, ce travail nous offre tout de même un éclairage sur les processus de pensée qui pourraient se retrouver dans notre caractérisation.

Une autre caractérisation qui ne tient pas compte du type de caractéristiques sélectionnées est celle de Weintrop et de son équipe (2016) qui ont proposé une taxonomie de la pensée computationnelle pour l'enseignement des sciences et des mathématiques. La taxonomie était divisée en quatre catégories. Ces catégories mettent de l'avant différents aspects de la pensée computationnelle et nous éclairerons dans notre travail, mais elles ne sont pas propres à l'activité mathématique et ne distinguent pas les idées qui relèvent du raisonnement, de celles qui relèvent des concepts ou des représentations. Deux des catégories sont liées à des activités (résolution de problème et modélisation), alors que l'une d'elles semble se référer aux représentations et aux manières d'opérer et de raisonner sur celles-ci. La quatrième catégorie se réfère aux pratiques de données (collecte, création, manipulation, analyse, visualisation) en faisant appel à des activités, des concepts, des raisonnements et des représentations.

Les deux exemples précédents montrent qu'en distinguant trois dimensions, notre caractérisation sera davantage opératoire et facilitera notre travail pour analyser des activités algorithmiques en mathématique par la suite.

Conclusion

En conclusion, nous avons présenté la place qu'a eu l'activité algorithmique dans l'activité mathématique au fil du temps et la place qu'elle continue d'avoir. Nous avons ensuite défini la pensée algorithmique en mathématique, en nous appuyant sur la TO, comme une manière d'agir et de réfléchir dans des activités mathématiques faisant intervenir au moins un algorithme. Puis, nous avons présenté trois activités algorithmiques essentielles, la mobilisation, la conception et l'étude d'un algorithme, ainsi que les concepts, les raisonnements et les modes de représentation qui permettent de décrire la pensée algorithmique. Finalement, nous avons mis en lumière l'originalité de notre travail par sa manière de considérer la pensée algorithmique comme une forme de pensée mathématique, d'approcher la pensée algorithmique par l'activité algorithmique et de la caractériser selon trois dimensions. Nous poursuivrons notre travail en détaillant notamment chacune des classes d'activités selon les trois dimensions opératoires de la pensée algorithmique en mathématique présentées.

Références

Bråting, K., et Kilhamn, C. (2021). Exploring the intersection of algebraic and computational thinking. *Mathematical Thinking and Learning*, *23*(2), 170-185. https://doi.org/10.1080/10986065.2020.1779012

Brennan, K. et Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. *Proceeding of the Annual american educational research association meeting 1*, 85-90.

Broaley, L. (2015). *La programmation informatique dans la recherche et la formation en mathématiques au niveau universitaire* [mémoire de maitrise, Université de Montréal]. Papyrus. https://papyrus.bib.umontreal.ca/xmlui/handle/1866/12574?locale-attribute=fr

Denning, P. J. (2017). Remaining trouble spots with computational thinking. *Communications of the ACM, 60*(6), 33-39. https://doi.org/10.1145/2998438

Kallia, M., van Borkulo, S. P., Drijvers, P., Barendsen, E. et Tolboom, J. (2021). Characterising computational thinking in mathematics education: a literature-informed Delphi study. *Research in Mathematics Education, Informa UK Limited, 1-29*. http://doi.org/10.1080/14794802.2020.1852104

Knuth, D. E. (1974). Computer programming as an art. *Communications of the ACM, 17*(12), 667–673. https://doi.org/10.1145/361604.361612

Knuth, D.E. (1985). Algorithmic thinking and mathematical thinking. *The American Mathematical Monthly, 92*(3), 170-181. https://doi.org/10.2307/2322871

Modeste, S. (2012). *Teaching algorithm, what for? Some new questions for mathematics and issues for proof learning*. Centre pour la communication scientifique directe. https://tel.archives-ouvertes.fr/tel-00783294

Radford, L. (2011). Vers une théorie socioculturelle de l'enseignement-apprentissage : la théorie de l'objectivation. *Éléments, (1)*, 1-27.

Radford, L. (2015). Pensée mathématique du point de vue de la théorie de l'objectivation. Dans L. Theis (dir.), *Pluralités culturelles et universalité des mathématiques : enjeux et perspectives pour leur enseignement et leur apprentissage. Acte du colloque EMF2015* (p. 334-335). Université des Sciences et de la Technologie Houari Boumediene.

Robert, V. (2018). *Le développement de la pensée fonctionnelle dans les manuels scolaires du 3e cycle du primaire québécois : une analyse praxéologique* [thèse de doctorat, Université de Sherbrooke]. Savoirs UdeS. https://savoirs.usherbrooke.ca/handle/11143/12608

Ross, A. (2019). Algorithmes au cours de l'histoire. *Accromath, 14*(2).

Squalli, H. (2000). *Une reconceptualisation du curriculum d'algèbre dans l'éducation de base* [thèse de doctorat inédite]. Université Laval.

Squalli, H. (2015). La généralisation algébrique comme abstraction d'invariants essentiels. Dans L. Theis (dir.), *Pluralités culturelles et universalité des mathématiques : enjeux et perspectives pour leur enseignement et leur apprentissage. Acte du colloque EMF2015* (p. 334-335). Université des Sciences et de la Technologie Houari Boumediene.

Squalli, H. (2020). Early algebra : genèse d'un domaine de recherche, évolution et perspectives. Dans H. Squalli, I. Oliveira, A. Bronner et M. Larguier (dir.), *Le développement de la pensée algébrique à l'école primaire et au début du secondaire. Recherches et perspectives curriculaires* (p. 5-21). Livres en ligne du CRIRES. https://lel.crires.ulaval.ca/oeuvre/le-developpement-de-la-pensee-algebrique-lecole-primaire-et-au-debut-du-secondaire-recherches

Shute, V. J., Sun, C. et Asbell-Clarke, J. (2017). Demystifying computational thinking. *Educational Research Review, 22*, 142-158. http://doi.10.1016/j.edurev.2017.09.003

Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L. et Wilensky, U. (2016). Defining computational thinking for mathematics and science classrooms. *Journal of Science Education and Technology*, 25(1), 127-47. http://dx.doi.org/10.1007/s10956-015-9581-5

Wing, J. M. (2010). Computational thinking: What and why? *The Link*, 1-6. https://www.semanticscholar.org/paper/Computational-Thinking%3A-What-and-Why-Wing/d0dd04ea551f25af5ce4db87618ea386cb726195