

# Du « faire » au « dire » : le cas de la programmation par blocs

Marielle Léonard<sup>1,2</sup>

<sup>1</sup> Univ. Lille, CNRS, Centrale Lille, UMR 9189 CRIStAL, F-59000 Lille, France

<sup>2</sup> Univ. Lille, ULR 4354 - CIREL - Centre Interuniversitaire de Recherche en Éducation de Lille, F-59000 Lille, France

**Résumé.** Cet article investigate le passage d'une activité de programmation par blocs à la formulation explicite des concepts algorithmiques mobilisés. Dix-sept sujets de 9 à 15 ans ont enchaîné la programmation d'un robot virtuel sur une grille avec une tâche de catégorisation de grilles similaires à celles rencontrées précédemment.

L'étude des critères formulés pour grouper les grilles permet d'une part de dégager quatre types de critères, qui correspondent à des paliers dans la maîtrise des concepts algorithmiques sous-jacents. D'autre part, l'analyse montre que les termes employés pour désigner les concepts algorithmiques sont importants à considérer. Elle nous amène à pointer un besoin de vigilance à ce que les concepts soient désignés avec les termes adéquats, afin de ménager la progression ultérieure dans la conceptualisation.

**Mots-clés** : Programmation par blocs · Pensée informatique · Conceptualisation · Catégorisation

## 1 Introduction

Programmer un robot virtuel sur une grille est une tâche courante lors de l'initiation à la programmation informatique en langage par blocs, telle qu'elle est préconisée dans les programmes scolaires des cycles 3 et 4 depuis 2016<sup>3</sup>. La consigne est souvent formulée ainsi : « Programmez le robot pour qu'il [remplisse une certaine mission sur la grille] ». Lors de la résolution de la tâche, le sujet manipule des blocs qui sont les éléments du langage de programmation (Figure 1).

Lorsque le sujet réussit la tâche, c'est qu'il a utilisé les blocs mis à sa disposition pour concevoir un programme valide, sans avoir à formuler explicitement les concepts algorithmiques sous-jacents. Cependant, les grilles sont construites de telle manière que la solution experte mobilise un ou plusieurs concepts algorithmiques (répétition via le bloc répéter, structure conditionnelle via le bloc si/sinon, variable, fonction). Une solution alternative

---

3. BOEN n° 16 du 17 avril 2025, BOEN n° 31 du 30 juillet, BOEN n°11 du 26 novembre 2015

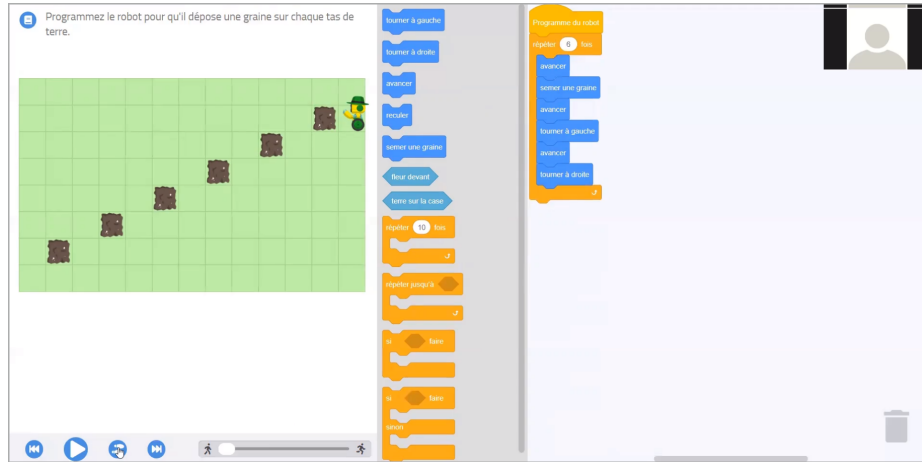


FIGURE 1 – Capture d’un enregistrement vidéo d’écran lors d’une tâche de programmation, le sujet étant en partage d’écran avec l’expérimentatrice

est parfois possible mais elle implique souvent un programme long et fastidieux à concevoir.

La programmation joue un rôle privilégié dans le développement de la pensée informatique, qui comprend un ensemble d’habiletés impliquées dans la résolution de problèmes et la compréhension de phénomènes et de systèmes : pensée algorithmique, processus de décomposition, de généralisation, d’abstraction et d’évaluation [1, 5].

Dans cet article, nous cherchons à approfondir ce qui se passe du point de vue de la conceptualisation lorsque les sujets sont soumis à des tâches de programmation d’un robot sur une grille de manière régulière. Dans quelle mesure cette activité réalisée de manière récurrente amène de jeunes sujets de 9 à 15 ans à repérer les structures algorithmiques sous-jacentes à la construction des grilles et à les formuler verbalement? Quels sont les indicateurs d’une conceptualisation des structures algorithmiques rencontrées?

## 2 Cadre théorique

### 2.1 Deux paliers dans le développement conceptuel

Le présent travail s’inscrit dans une approche constructiviste et cognitiviste de l’apprentissage. Il s’appuie sur le cadre théorique de Vergnaud relatif à la conceptualisation qui est définie comme « un moyen de réduire l’information à ce qui est nécessaire et suffisant pour comprendre et traiter une situation d’une certaine classe. » [7]. Vergnaud distingue deux paliers dans la conceptualisation. Un premier palier consiste en une conceptualisation-en-acte, c’est-à-dire intimement liée au déroulement de l’action dans les situations

auxquelles le sujet est confronté. Vergnaud associe la conceptualisation-en-acte à une forme opératoire de la connaissance, qui est « mise en œuvre dans l'action en situation » [8], qui « permet de faire et de réussir » [7]. Ces connaissances-en-acte sont constituées de schèmes (organisation invariante de l'action pour une classe de situations donnée), et reposent sur des invariants opératoires. Elles restent largement implicites, y compris pour le sujet lui-même. Le second palier dans la conceptualisation est caractérisé par la forme prédicative de la connaissance, qui prend la forme d'énoncés et de textes pour exprimer les relations entre les objets, leurs propriétés, leurs transformations [8].

Selon Vergnaud, la désignation verbale des objets et de leurs propriétés est essentielle dans le processus de conceptualisation, mais elle s'avère difficile, et ce d'autant plus que les sujets sont jeunes. Accéder à une forme prédicative de la connaissance sous-tend un processus de généralisation, notamment à travers une opération linguistique de nominalisation [7]. C'est pourquoi, il est observé de manière générale un décalage entre les connaissances-en-acte, ce que le sujet est capable de faire, et leur formulation, ce qu'il est capable d'énoncer ou d'expliquer [8].

L'objet de cet article est d'étudier le passage entre ces deux formes de conceptualisation dans le cas de la programmation par blocs, qui s'inscrit en premier lieu dans l'action. Quelles sont les caractéristiques du passage de la conceptualisation-en-acte, observable à travers l'activité du sujet mais restant implicite, à une conceptualisation explicite, verbalisée, de concepts algorithmiques ?

## 2.2 La catégorisation comme indicateur du développement conceptuel

Pour Hofstadter et Sander, la catégorisation est une opération mentale qui consiste à assigner une certaine entité à une catégorie, une catégorie étant définie comme une « structure mentale qui permet d'organiser les informations et d'y accéder. » [3]. La catégorisation agit comme un filtre. En ignorant certaines propriétés et en en rendant d'autres saillantes, elle révèle un point de vue sur une situation. Les catégorisations admettent des critères qui vont du concret vers l'abstrait, du spécifique vers le général. Les critères retenus par les sujets pour catégoriser des situations est un indicateur de leur niveau d'expertise par rapport à celles-ci. Pour les novices, les éléments saillants, qui orientent le traitement de la situation, sont essentiellement les traits de surface, qui concernent la structure sémantique de la situation, son habillage, et sont inutiles dans le traitement de celle-ci. Par contraste, ce sont les traits de structure qui sont privilégiés par les experts, c'est-à-dire les propriétés utiles pour le traitement de la classe de situations [3, 4].

La flexibilité conceptuelle, qui est caractérisée par le passage d'une catégorisation à une autre, est un indicateur du développement conceptuel. Cette flexibilité est le plus souvent induite par un changement contextuel. L'évolution de la catégorisation intervient spontanément lors de la

confrontation à des situations nouvelles, ou lors de situations spécifiquement construites dans cette optique (contexte scolaire) [4]. Pour Sander, dans une perspective développementale de l'apprentissage, l'enjeu est donc de favoriser le développement de catégories qui s'activent dans les contextes appropriés à leur usage. En effet, le développement et la mobilisation de catégories mentales conditionnent les acquisitions de connaissances. Ces catégories sont porteuses d'inférences, chaque situation bénéficiant des connaissances associées à la catégorie dans laquelle le sujet la range.

### 3 Cadre méthodologique

Nous nous appuyons sur les éléments théoriques précédents pour construire une tâche de catégorisation dans notre contexte de l'initiation à la pensée informatique. L'objectif de cette tâche de catégorisation est d'amener les sujets à verbaliser les connaissances acquises lors des situations de programmation auxquelles ils ont été confrontés précédemment.

Tijus et Cordier ont réalisé une description formelle des tâches de catégorisation et ont relevé les méthodes expérimentales associées, utilisées en psychologie cognitive et psychologie du développement. Notamment, une catégorie peut être définie en extension, par l'énumération de ses éléments, ou par compréhension, par la description de propriétés qui la caractérisent [6].

Dans l'arborescence de tâches de catégorisation construite par ces auteurs, nous relevons la tâche qui consiste à présenter un groupe d'objets, puis à demander au sujet de « mettre ensemble ce qui va ensemble », de relever les regroupements qu'il effectue spontanément, puis de lui demander de justifier son choix, soit en dénommant la catégorie créée, soit en décrivant les propriétés. Tijus et Cordier indiquent, qu'en faisant varier le degré de similarité entre les objets présentés, cette tâche permet d'étudier le processus de construction de catégories.

En prenant appui sur ces travaux, nous mettons en place une tâche de catégorisation de grilles similaires à celles déjà rencontrées lors de l'activité de programmation proprement dite. Notre objectif, à travers cette tâche, est d'amener le sujet à chercher des liens entre les situations de programmation, à les organiser les unes par rapport aux autres et à verbaliser cette organisation. Nous présentons au sujet des planches de six grilles identiques ou similaires à celles qu'il a rencontrées lors des tâches de programmation (figure 2). Nous lui demandons quelles grilles il rangerait ensemble et d'expliquer ses regroupements. En d'autres termes, nous lui demandons de créer des catégories, et de les définir d'une part en extension en listant les grilles qu'il range dedans, et d'autre part en compréhension lorsque nous lui demandons de justifier son choix.



FIGURE 2 – Planche 1 : première planche de 6 grilles présentée aux sujets lors de la tâche de catégorisation

## 4 Protocole expérimental

L'expérimentation comporte deux temps distincts, qui visent à séparer le moment où on évalue les connaissances-en-acte en programmation, de celui où on évalue les connaissances prédicatives.

La première phase, de 45 minutes, comprend des tâches de programmation d'un robot virtuel sur une grille (Figure 1). Lors de cette première phase, le sujet traite les situations de programmation les unes après les autres, sans que nous lui demandions explicitement d'établir de liens entre elles. Tous les programmes exécutés sont enregistrés pour aider à repérer les connaissances-en-acte.

Cette première phase est suivie de la tâche de catégorisation de grilles, qui constitue le cœur du présent article. Nous avons construit 13 planches, en fonction de critères que nous avons anticipé a priori (contexte sémantique, système d'orientation, parcours du robot, notions algorithmiques en jeu). La première planche que nous présentons systématiquement aux sujets est celle de la figure 2. Ensuite, l'ordre de présentation des planches dépend des critères de catégorisation avancés. Dans la mesure du possible, lorsqu'un critère est formulé par le sujet, il ne peut plus s'appliquer pour distinguer les grilles de la planche suivante, afin d'inciter le sujet à élargir ses critères de catégorisation. Par exemple, si un sujet groupe les grilles A et E de la planche 1 parce que les éléments sont alignés, les grilles C et F parce qu'ils sont en diagonale, et B et D parce qu'ils sont disposés autrement, alors nous lui présentons ensuite la planche 6 (Figure 4) où les éléments sont alignés sur toutes les grilles.

L'expérimentation s'est déroulée pendant l'été 2022, en visioconférence, avec enregistrement de la voix. Elle a impliqué 17 sujets, scolarisés du CE2 à la

3<sup>ème</sup>. Ceux-ci ont participé précédemment, pendant entre un et deux ans, à des sessions du concours de programmation en ligne Algoréa<sup>4</sup>, en visioconférence, dans le cadre d'une étude plus large. Ils sont donc familiers de l'interface de programmation, des tâches de programmation de robot sur une grille, du format de visioconférence, et la confiance avec l'expérimentatrice est établie. Par ailleurs, quinze de ces sujets (dont les prénoms ont été changés) sont identifiés comme scolairement bons élèves, deux (Alban et Robin) sont repérés comme plus fragiles, sans toutefois être en difficulté scolaire.

## 5 Résultats et analyse

Chaque sujet a catégorisé les grilles d'une dizaine de planches, ce qui représente un corpus de 168 catégorisations pour l'ensemble des 17 sujets. Après transcription des verbatims, nous obtenons un document de 49600 mots.

La tâche de catégorisation proposée s'est avérée loin d'être évidente. En réponse à la consigne « Mettre ensemble ce qui va ensemble », la stratégie des sujets a consisté à chercher des points communs entre les différentes grilles sans se soucier que toutes les grilles soient prises en compte dans la catégorisation, que l'intersection entre les groupes constitués soit non vide, ou que les critères avancés soient de même nature. Ils aboutissent donc souvent à des catégorisations qui ne sont pas des partitions de l'ensemble des grilles de la planche. Par ailleurs, certains sujets proposent spontanément plusieurs catégorisations pour une même planche, révélant ainsi soit une bonne flexibilité conceptuelle, soit une difficulté à se fixer sur des critères cohérents entre eux.

Ces éléments expliquent que nous obtenions beaucoup plus de critères formulés (318) que de catégorisations (168).

### 5.1 Nature des critères formulés

Après une première exploration des données, nous regroupons les critères évoqués selon leur nature, en quatre catégories : descriptive, géométrique, dynamique et algorithmique.

*Critères de nature descriptive - 71 occurrences* - Nous regroupons les traits de surface, qui correspondent à la structure sémantique du problème [4] dans cette catégorie : description de la scène, de son décor, aspect du robot, taille de la grille, présence de certains éléments sur la grille, présence de tests lorsqu'ils sont évoqués isolément comme une caractéristique visuelle des grilles.

*Critères de nature spatiale - 64 occurrences* - Nous regroupons l'identification de dispositions particulières des éléments sur la grille (ligne droite, escalier, croix,...), de configurations géométriques (diagonale, carré...), et la description d'espacements, réguliers ou non, entre les éléments de la grille.

---

4. <https://algorea.org>

*Critères de nature dynamique - 68 occurrences* - Les critères de nature dynamique comprennent la description du parcours du robot, les actions que le robot doit réaliser lorsqu'elles sont regroupées selon leur type (collecter/déposer/atteindre une case cible), l'évocation des instructions de déplacement dans les différents systèmes d'orientation (avancer/ pivoter à droite/gauche vs avancer vers le nord/ sud/est/ouest). Ces critères révèlent que le sujet anticipe, avec un certain degré de généralisation, la séquentialité des actions et déplacements du robot virtuel.

*Critères de nature algorithmique - 115 occurrences* - Nous regroupons les formulations de concepts algorithmiques sous-jacents à la construction des grilles : répétition, structure conditionnelle, fonction.

Prénom	Niveau scolaire	Score Programm.	Connaissances en acte				Nature des critères					
							descriptive	spatiale	dynamique	algorithmique		
Axel	5 <sup>ème</sup>	200	B1	B2	B3	C	V	F	0	0	2	12
Louna	6 <sup>ème</sup>	200	B1	B2	B3	C	V	F	0	3	2	13
Alexis	6 <sup>ème</sup>	200	B1	B2	B3	C	V	F	0	8	2	16
Mickaël	4 <sup>ème</sup>	190	B1	B2	B3	C	V	F	2	1	2	15
Timéo	5 <sup>ème</sup>	180	B1	B2	B3	C	V	F	0	4	2	19
Noé	3 <sup>ème</sup>	180	B1	B2	B3	C	V	F	0	2	2	15
Sam	6 <sup>ème</sup>	150	B1	B2	B3	C			0	5	7	11
Louane	CM2	150	B1	B2	B3	C			6	2	6	1
Lou	6 <sup>ème</sup>	140	B1	B2	B3	C	F		8	3	9	2
Colin	6 <sup>ème</sup>	130	B1	B2	B3	C	V		5	6	15	7
Alban	CM2	120	B1	B2		C			12	1	1	0
Ugo	CM1	110	B1	B2	B3	C			5	8	8	0
Léa	CE2	100	B1	B2	B3	C			10	5	2	2
Alix	CM2	100	B1	B2	B3				5	5	3	0
Élana	CM1	90	B1	B2	B3				3	1	3	1
Robin	CM1	70	B1	B2	B3				5	7	2	1
Paula	CE2	60	B1						10	3	0	0
TOTAL									71	64	68	115

FIGURE 3 – Confrontation entre scores au parcours de programmation, connaissances-en-acte et nature des critères formulés lors de la tâche de catégorisation

Sur la figure 3, nous confrontons les critères de catégorisation verbalisés par les sujets avec leur score aux tâches de programmation réalisées juste avant et les connaissances-en-acte révélées par les programmes conçus. Plus précisément, la figure 3 montre, pour chaque sujet :

- le score obtenu au parcours de programmation, sur 200 points
- les concepts algorithmiques mobilisés, avec les abréviations suivantes :
  - **B1** Séquence d'instructions et répétition d'une seule instruction
  - **B2** Répétition d'une séquence d'instructions
  - **B3** Répétition utilisée au sein de structures plus complexes (imbrication, répétitions conditionnelles)

- **C** Structure conditionnelle
- **V** Variable
- **F** Décomposition du programme en utilisant une ou plusieurs fonctions

Les concepts mentionnés sont ceux qui ont été mobilisés, ceux en gras sont ceux dont l'utilisation aboutit à un programme valide pour toutes les tâches où ils ont été utilisés (parfois après plusieurs essais), donc qui ont fait l'objet d'une conceptualisation-en-acte.

- le nombre d'occurrences pour chaque nature de critères lors de la tâche de catégorisation.

Nous constatons un lien très net entre un score élevé au parcours de programmation, une connaissance-en-acte des concepts algorithmiques sous-jacents et la formulation de critères algorithmiques lors de la tâche de catégorisation. Tous les sujets qui maîtrisent l'ensemble des concepts algorithmiques de base mobilisent ces traits de structure lors de la tâche de catégorisation. Plusieurs structures algorithmiques sont souvent évoquées comme titres de groupements lors de la même catégorisation. Ainsi, sur les 55 mentions de la structure répétitive et les 44 mentions de la structure conditionnelle, 32 sont mentionnées dans la même catégorisation. Lorsqu'un critère algorithmique ne permet pas de différencier les grilles, ces sujets experts proposent majoritairement en première intention de les mettre toutes ensemble, avant de se rabattre sur une autre nature de critère, spatiale ou dynamique. Ils ne reviennent pas sur des critères descriptifs (à une exception près).

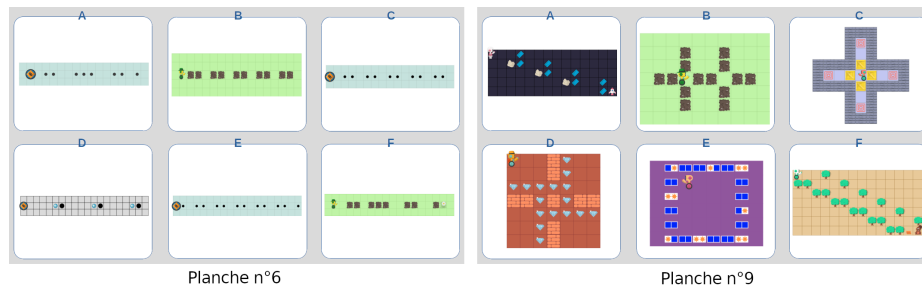


FIGURE 4 – Exemples de planches présentées aux sujets lors de la tâche de catégorisation

Par exemple, Louna repère et met ensemble les grilles avec une répétition sous-jacente sur la planche 6 (Figure 4). Pour les deux grilles restantes, elle considère la disposition irrégulière des éléments, qu'elle qualifie d'« aléatoire », ce qui correspond à une structure conditionnelle imbriquée dans une boucle, structure qu'elle a pourtant conceptualisée en acte, comme en témoigne son score au parcours de programmation, mais qu'elle n'arrive pas à formuler. Nous

retrouvons le décalage entre conceptualisation-en-acte et conceptualisation sous sa forme verbalisée pointé par Vergnaud [7].

Les critères de nature spatiale et de nature dynamique peuvent être qualifiés d’intermédiaires entre les critères descriptifs, traits de surface selon Hofstadter et Sander, et les critères de nature algorithmique, qui sont des traits de structure. Tous les sujets ont proposé des critères soit de nature spatiale, soit de nature dynamique. À l’exception d’Alban et Paula, pour qui les critères descriptifs sont vraiment majoritaires, les sujets proposent plutôt des critères intermédiaires en première intention et reviennent à des critères descriptifs comme « critères refuges », lorsqu’ils ne trouvent plus de critères intermédiaires, notamment pour former un groupe avec les grilles non encore prises en compte. Les critères spatiaux et dynamiques sont explicitement associés dans certains cas, ce qui marque une progression de la conceptualisation vers les critères de nature algorithmique. Par exemple, sur la planche 9 (Figure 4), Alexis (planche 9) associe explicitement la configuration des éléments de la grille avec la planification temporelle du parcours du robot, exprimant une sorte d’équivalence entre les deux. La structure algorithmique (répétition), qu’il maîtrise pourtant parfaitement, reste implicite car découlant de la disposition des éléments sur la grille. C’est le dialogue avec l’expérimentatrice qui amène Alexis à verbaliser le lien avec la structure algorithmique.

Le passage d’une catégorisation à une autre est encouragée par le protocole expérimental dans la mesure où il consiste à neutraliser, d’une planche à l’autre, les critères de catégorisation avancés par les sujets, favorisant ainsi la flexibilité conceptuelle. Hormis le passage à des critères algorithmiques déjà évoqué pour les plus experts, nous identifions un autre palier dans la conceptualisation, lorsque les sujets ont une bonne connaissance-en-acte de la répétition et une première maîtrise de la structure conditionnelle (Louane, Lou, Colin, Ugo sur la figure 3). Nous observons à ce moment une évolution d’une catégorisation selon des critères descriptifs et spatiaux à une prise en compte plus fréquente de l’aspect séquentiel de la programmation (critères de nature dynamique).

Ainsi, dans une approche s’appuyant sur la flexibilité au sens de Sander, les critères de nature spatiale et ceux de nature dynamique tiennent un rôle important dans la progression de la catégorisation vers des critères de structure.

## 5.2 Nature grammaticale des termes employés

Afin d’affiner notre analyse, nous investiguons, de manière complémentaire, la nature grammaticale des termes que les sujets emploient pour formuler leurs critères. Nous présentons cette analyse pour les critères de nature algorithmique, et de manière plus précise pour la répétition (figure 5).

Pour la structure algorithmique de répétition, nous relevons 123 mentions, issues de 53 des 168 catégorisations. Elle est évoquée par 12 des 17 sujets, mais pas de la même manière. Les sujets les moins avancés décrivent des exemples et utilisent le verbe « répéter » (« Il faut répéter », « On répète », « Ça c’est répété »), alors que les plus experts utilisent en plus des noms (« boucle »,

Prénom	Niveau scolaire	Score Programm.	Connaissances en acte	Nb planches	Occurrences verbe répéter	Noms employés :
						- occurrences - terme le plus fréquent
Axel	5 <sup>ème</sup>	200	B1 B2 B3 C V F	5	0	« des patterns » (3) 6
Louna	6 <sup>ème</sup>	200	B1 B2 B3 C V F	7	2	« boucle » (6) 8
Alexis	6 <sup>ème</sup>	200	B1 B2 B3 C V F	6	10	« répétition » (3) 5
Mickaël	4 <sup>ème</sup>	190	B1 B2 B3 C V F	7	3	« séquence répétée » (14) 28
Timéo	5 <sup>ème</sup>	180	B1 B2 B3 C V F	8	9	« répétition » (8) 9
Noé	3 <sup>ème</sup>	180	B1 B2 B3 C V F	5	4	« le bloc répéter » (5) 7
Sam	6 <sup>ème</sup>	150	B1 B2 B3 C	9	9	« algorithme » (7) 9
Louane	CM2	150	B1 B2 B3 C	1	0	« le bloc répéter » 4
Lou	6 <sup>ème</sup>	140	B1 B2 B3 C F	1	4	« des répéter dix fois » 1
Colin	6 <sup>ème</sup>	130	B1 B2 B3 C V	3	5	0
Alban	CM2	120	B1 B2 C	0	0	0
Ugo	CM1	110	B1 B2 B3 C	0	0	0
Léa	CE2	100	B1 B2 B3 C	2	2	0
Alix	CM2	100	B1 B2 B3	0	0	0
Élana	CM1	90	B1 B2 B3	1	1	0
Robin	CM1	70	B1 B2 B3	0	0	0
Paula	CE2	60	B1	0	0	0

FIGURE 5 – Termes employés par les sujets pour évoquer le concept de répétition lors de la tâche de catégorisation de grilles

« répétition » mais aussi « patterns », « bloc répéter » ou « algorithme »). L'emploi de substantif (« le répéter ») constitue une étape intermédiaire entre l'emploi du verbe et l'émergence d'une désignation du concept par un nom. Dans certains cas, le concept est désigné par le nom de l'élément de langage (« le bloc répéter »). Cette dénomination témoigne d'une conceptualisation de la répétition qui reste contextuelle au langage par blocs. La question d'une difficulté au moment de la transition vers un langage textuel se pose. Nous trouvons aussi des cas où la structure algorithmique est identifiée comme critère de groupement des grilles en la désignant par un nom, mais le terme employé pour la désigner est erroné. C'est le cas par exemple de Sam qui exprime le concept de répétition en parlant d'« algorithme ».

Ces désignations approximatives ou erronées sont encore plus fréquentes pour les concepts de structure conditionnelle et de fonction. Concernant la structure conditionnelle, nous relevons les termes « capteur », « détecteur », « condition », « un si », « un si sinon ». Aucun de ces termes ne désigne le concept de manière adéquate. Pour le concept de fonction, outre le terme « fonction », nous trouvons « le truc violet » qui fait référence à l'apparence du bloc en Scratch, mais aussi le terme « variable », qui révèle une confusion latente entre les concepts. Ainsi, nous avons de nombreuses occurrences où une conceptualisation est verbalisée, alors que les termes dédiés pour nommer le concept ne sont pas du tout maîtrisés.

## 6 Conclusion et perspectives

Au cours de l'analyse précédente, nous avons montré que la conceptualisation des structures algorithmiques de base est nettement corrélée avec les connaissances-en-acte acquises au cours de l'activité de programmation d'un robot virtuel sur une grille. À travers une tâche de catégorisation de grilles effectuée aussitôt après les tâches de programmation, nous avons repéré deux indicateurs qui caractérisent le développement conceptuel dans ce contexte.

Un premier indicateur concerne la nature des critères proposés pour regrouper les grilles, critères qui vont des plus concrets vers les plus abstraits, marquant une progression dans la conceptualisation, avec deux natures intermédiaires qui réfèrent aux aspects spatiaux et temporels. Dans le cadre de la programmation d'un robot sur une grille, ceux-ci doivent être mis en correspondance, ce qui constitue un pré-requis à l'émergence de critères de nature algorithmique. La modélisation de la figure 6 en montre une synthèse.

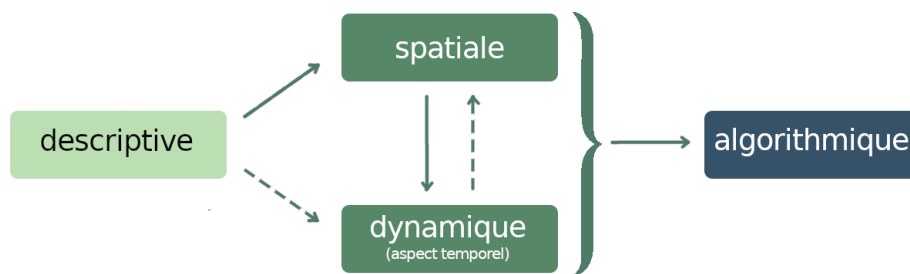


FIGURE 6 – Progression dans la nature des critères formulés lors de la tâche de catégorisation de grilles (la flèche pointillée indique un passage possible mais moins fréquent)

Le second indicateur est relatif à la nature grammaticale des termes employés pour formuler un concept, la maîtrise experte étant caractérisée par la désignation de ce concept par un nom (vs par un verbe).

Ces deux indicateurs sont cohérents avec les cadres théoriques que nous avons mobilisés, dans lesquels ils sont déjà identifiés dans des contextes plus généraux : travaux de Vergnaud sur la conceptualisation, travaux de Sander sur le développement conceptuel. Nous avons retrouvé d'une part un décalage entre une conceptualisation-en-acte et la formulation des concepts, ainsi que le processus de nominalisation [7, 8]. D'autre part, les critères de catégorisation que nous avons identifiés sont bien progressifs, des plus concrets vers les plus abstraits [4].

Cette étude présente toutefois quelques limites que nous pointons, mais qui constituent un appui pour des expérimentations ultérieures. Outre la taille restreinte de l'échantillon, nous constatons que les résultats des sujets sont

largement en lien avec leur niveau scolaire, puisque ce sont les sujets les plus âgés qui obtiennent à la fois un score élevé aux tâches de programmation et mobilisent le plus souvent des critères algorithmiques pour catégoriser les grilles. Une perspective consiste à refaire une expérimentation similaire sur un échantillon plus large, avec des sujets de niveaux scolaires proches. Les niveaux CM2 et 6<sup>ème</sup> semblent être appropriés car ils constituent des niveaux charnières, avec une forte variabilité, dans le cas de notre échantillon limité. Comme l'analyse d'un plus grand volume de verbatims deviendrait irraisonnable à la main, nous visons d'automatiser la catégorisation des verbatims. Nous avons déjà construit une ébauche de modèle intégrant un LLM dans cette optique.

Dans le cadre de cette expérimentation hors temps scolaire, les sujets ont seulement réalisé les tâches proposées, aucune phase de structuration des concepts n'a été organisée. Nous constatons la désignation de concepts sans être en mesure d'identifier dans quel contexte (scolaire? familial?) les sujets ont acquis ces termes. Cependant, nous soulignons l'importance de ne pas se contenter d'une conceptualisation-en-acte, même si celle-ci constitue une première étape nécessaire. Afin de favoriser la progression des élèves dans la conceptualisation, il est essentiel de pointer et de nommer précisément les concepts algorithmiques sous-jacents aux tâches de programmation, dans le cadre d'un enseignement structuré de la programmation par blocs, ce qui est encore loin d'être le cas [2]. De plus, il nous semblerait pertinent d'introduire les deux structures de contrôle aussi tôt qu'il est possible, afin de favoriser la distinction des cas d'usage.

## Bibliographie

- [1] Csizmadia, A., Curzon, P., Dorling, M., Humphreys, S., Ng, T., Selby, C., Woollard, J. : Computational thinking. a guide for teachers. Computing at school (2015)
- [2] Gaubert-Macon, C., Chesneaux, J.M., Desprez, J.M., Picaronny, C., Montreuil, V. : Pratique de l'informatique aux cycles 3 et 4 (2022), URL <https://www.education.gouv.fr/pratique-de-l-informatique-aux-cycles-3-et-4-344254>
- [3] Hofstadter, D., Sander, E. : L'analogie, cœur de la pensée. Odile Jacob (2013), URL <https://hal.science/hal-02113546/>
- [4] Sander, E. : Des catégories pour raisonner : une perspective fondée sur la flexibilité conceptuelle. In : Fonctions exécutives, apprentissages et réussite scolaire, pp. 97–118, Dunod (2025), URL [https://shs.cairn.info/article/DUNOD\\_DUROI\\_2025\\_01\\_0097](https://shs.cairn.info/article/DUNOD_DUROI_2025_01_0097)

- [5] Selby, C., Woollard, J. : Computational thinking : the developing definition (2013)
- [6] Tijus, C.A., Cordier, F. : Psychologie de la connaissance des objets. catégories et propriétés, tâches et domaines d'investigation. L'année psychologique **103**(2), 223–256 (2003), <https://doi.org/10.3406/psy.2003.29635>
- [7] Vergnaud, G. : Forme opératoire et forme prédicative de la connaissance. Actes du Colloque GDM (2001)
- [8] Vergnaud, G. : La pensée est un geste. comment analyser la forme opératoire de la connaissance. Enfance **1**, 37–48 (2001), <https://doi.org/10.3917/enf1.111.0037>