

POUR UNE ANALYSE MULTI - CRITERES
D' ACTIVITES DE PROGRAMMATION EN LOGO

C. DUPUIS, M-A. EGRET, D.GUIN

L'analyse de l'activité de programmation ne peut se faire sans la définition de critères. Les auteurs définissent ces critères et les appliquent aux observations de l'apprentissage du Logo par des élèves de 13-15 ans.

1 . INTRODUCTION .

L'introduction de l'informatique dans l'enseignement , depuis quelques années déjà , a suscité de nombreux travaux de recherche en didactique , psychologie et sciences de l'éducation . On peut y distinguer deux domaines de recherche :

- l'informatique considérée comme une discipline ou l'informatique " **objet** " .
- l'informatique intervenant dans d'autres disciplines ou l'informatique " **outil** " .

Nous nous intéressons ici essentiellement à la première approche : notre étude est axée sur l'informatique "**objet**" et les problèmes que pose, aux élèves, l'apprentissage d'un langage de programmation .

L'analyse des activités de programmation comprend l'analyse a priori des situations-problèmes , de la tâche demandée à l'élève , mais aussi l'analyse a posteriori des stratégies de résolution des élèves . **La définition de critères d'analyse nous paraît indispensable pour la communication et la comparaison des recherches expérimentales menées dans un même domaine .**

Pour une analyse multi-critères d'activités
de programmation en Logo

“ Plus que les résultats de recherche, ce sont sans aucun doute les méthodes qui peuvent être les plus utiles pour les enseignants , non pas les méthodes de " preuves " des résultats mais plutôt les **méthodes d'analyse des situations** proposées aux élèves , à la fois lorsqu'il s'agit d'introduire des notions et quand il s'agit d'évaluer des acquisitions ” (J. Rogalski , 1986) .

Notre propos est de présenter ici , sur une situation et une classe d'élèves , les critères d'analyse que nous avons définis au cours de notre expérimentation . Nous travaillons avec une classe de 4^{ème} de 19 élèves (13-15 ans) . Les élèves suivent une heure hebdomadaire d'informatique (leur horaire de mathématique est de 4 heures par semaine) . Cette heure supplémentaire est assurée par leur professeur de mathématique . La première partie de l'expérimentation , qui est celle qui nous intéresse ici , a consisté en un enseignement de base du langage Logo , graphique et non graphique , en insistant sur la structuration des programmes .

Un compte - rendu détaillé se trouve dans la brochure Logo 3 . Programmation structurée : Présentation et Analyse de situations (C. Dupuis, M. - A. Egret , D. Guin , 1987) .

Cette étude s'intègre dans un ensemble de recherches de l'IREM de Strasbourg liées à l'introduction de l'informatique dans le système scolaire . Elle a été menée dans le cadre du GRECO " Didactique et Acquisition des Connaissances Scientifiques " du CNRS qui coordonne le travail de plusieurs équipes de recherche sur la Didactique des Mathématiques et de l'Informatique .

2 . POURQUOI DES CRITERES D'ANALYSE ?

2 . 1 Qu'est - ce qu'un critère d'analyse ?

L'idée de critère d'analyse , dans une recherche expérimentale en didactique , est proche de celle de variable qualitative avec le sens que lui donnent les statisticiens en analyse des données . Une variable qualitative à modalités est **une application de l'ensemble des individus dans l'ensemble des modalités** . Dans notre recherche , un individu est un programme écrit par un élève . Ce pourrait être aussi un ensemble de programmes écrits par le même élève . La définition de chaque variable - critère et de ses modalités induit une **partition de l'ensemble des individus** et permet donc d'explicitier, au - delà des différences apparentes ou superficielles , ce que certains programmes ont en commun .

2 . 2 Au - delà de la réussite ...

Il n'est guère raisonnable de se contenter d'observer seulement **la réussite ou l'échec** , sauf sur une tâche ponctuelle . D'une part , la réussite peut , en programmation comme en mathématique , être le résultat de stratégies fort différentes . D'autre part , l'échec peut avoir des causes diverses que les critères d'analyse doivent distinguer à partir des indicateurs que sont les productions des élèves .

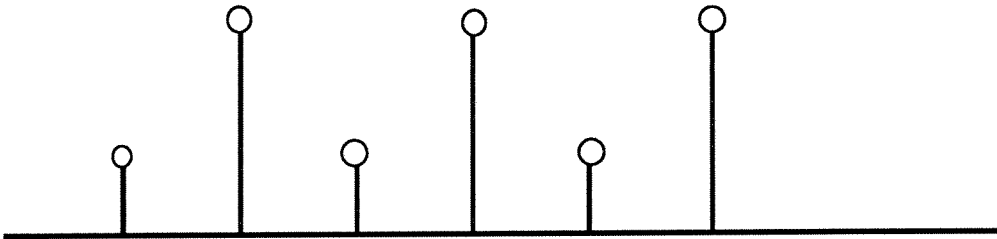
" **Les critères d'analyse ne sont pas fondés sur la proximité de la réponse avec le réalisme graphique de la réponse correcte , mais sur la nature des processus psychologiques en jeu inférée à partir des productions**" (P. Mendelsohn , 1985) .

2 . 3 Une analyse multi - critères .

La complexité de la tâche de programmation nécessite la définition de plusieurs critères d'analyse . Chaque critère qui intervient dans l'analyse reflète un des aspects de l'activité. Il faut veiller cependant , à **l'indépendance des critères** qui se traduit par la possibilité, a priori , d'associer n'importe quelle modalité d'un critère donné avec n'importe quelle modalité d'un autre critère pour constituer le **profil d'un individu** .

3 . LA SITUATION .

Nous illustrerons les critères en vous présentant un test individuel , baptisé BALLONS , réalisé au bout de seize heures de pratique active de la programmation . Ce test avait déjà été proposé à des élèves dans des conditions d'expérimentation différentes (J. Hillel , R. Samurçay , 1985) . Nous voulions comparer les résultats pour voir l'impact de notre enseignement de la programmation structurée . Les élèves ont pour consigne d'écrire le programme permettant de réaliser à l'écran un dessin semblable à celui-ci :



ATTENTION aux remarques suivantes :

- les distances entre les pieds des tiges sont égales .
- les cercles ont tous le même rayon .
- il y a deux longueurs de tiges .




Il est possible d'écrire un programme Logo permettant de réaliser à l'écran ce dessin en pilotant la tortue de façon à ce qu'elle suive fidèlement le tracé . Dans ce cas , on utilise seulement les instructions qui constituent le vocabulaire initial du langage Logo .

Pour une analyse multi-critères d'activités
de programmation en Logo

Ce type de programme n'utilise pas une des grandes richesses du langage Logo qui est la possibilité de structurer son programme en le décomposant en " procédures " que l'utilisateur définit lui - même : une **procédure** est une suite d'instructions associées par l'utilisateur à un mot qui devient alors le **nom** de la procédure . La réalisation du projet d'ensemble implique la **coordination** des procédures ainsi définies . Nous appelons **interface** une procédure (ou suite d'instructions) intermédiaire définie par l'utilisateur pour réaliser la coordination entre deux procédures .

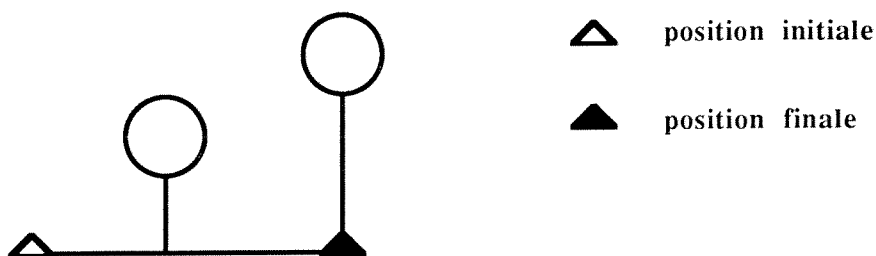
A titre d'exemple , voici un ensemble de procédures réalisant le projet BALLONS :

Pour illustrer notre propos , nous représenterons les procédures par leur exécution . Plus exactement , il s'agit d'un schéma d'exécution puisque nous faisons figurer les positions initiale et finale de la tortue alors qu'elles ne peuvent apparaître simultanément à l'écran. Dans les schémas d'exécution , nous représentons la tortue par un triangle permettant de visualiser sa position et son orientation :

-  tortue orientée vers le haut
(cap 0)
-  tortue orientée vers la droite
-  tortue orientée vers le bas

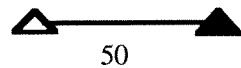
Le projet d'ensemble BALLONS peut être décomposé de la manière suivante :

DESSIN



Pour une analyse multi-critères d'activités
de programmation en Logo

DESSIN est répété trois fois et suivi de TRAIT :



▲ : position initiale

▲ : position finale

POUR TRAIT

TD 90

AV 50

TG 90

FIN

POUR BALLONS

REPETE 3 [DESSIN]

TRAIT

FIN

Le DESSIN lui - même est décomposé en deux procédures INTERFACE identiques , réalisant les déplacements horizontaux de la tortue et deux procédures BALLON : L dans lesquelles la longueur L de la tige ne sera pas la même .

POUR DESSIN

INTERFACE

BALLON 10

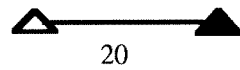
INTERFACE


BALLON 30


FIN

Pour une analyse multi-critères d'activités
de programmation en Logo

INTERFACE



 : position initiale

 : position finale

POUR INTERFACE

TD 90

AV 20

TG 90

FIN

POUR BALLON :L

AV :L

TD 90

POLY 2 36

TG 90


RE :L

FIN

exemple : BALLON 10



L = 10

 positions initiale et
finale confondues

Pour une analyse multi-critères d'activités
de programmation en Logo

La procédure POLY , qui permet d'obtenir un cercle , a un statut particulier car elle a été écrite au préalable dans un autre contexte (voir § 4.2.2.) .

POUR POLY : COTE : NB
REPETE : NB [AV : COTE TG 360 / : NB]
FIN

L' enseignement préalable était résolument orienté vers la structuration des programmes . Nous avons recherché des situations - problèmes où la structuration soit ressentie non comme une contrainte supplémentaire imposée par l'enseignant mais comme une stratégie économique de résolution du problème .

Dans le test BALLONS , la figure est présentée sans indication de positions initiale ou finale de la tortue ni de décomposition . Aucune consigne supplémentaire n'est donnée . Les choix des procédures de décomposition et de coordination sont donc à la charge de l'élève . Nous voulons observer les stratégies de programmation des élèves et savoir , parmi les outils de programmation mis à leur disposition dans l'enseignement , ceux qu'ils choisiront d'utiliser .

4 . DEFINITION DE SIX CRITERES D ' ANALYSE .

Nous aborderons successivement les stratégies de décomposition , le traitement de la coordination , l'utilisation des procédures POLYGONE pour tracer les cercles , le traitement de la répétition , la décomposition en sous - procédures et l'utilisation de variables .

4 . 1 Stratégies de décomposition : le critère ETAT .

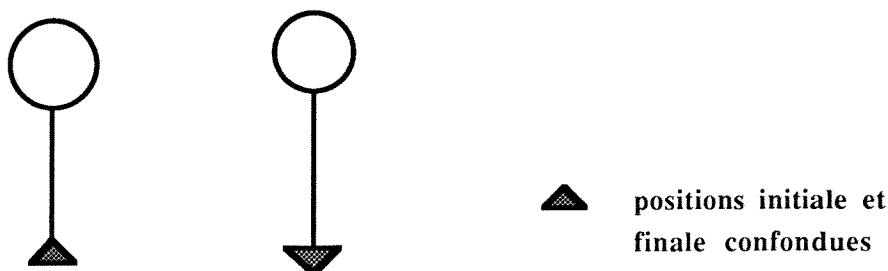
L'état de la tortue se caractérise par sa position et son orientation . Nous distinguerons ici trois types de **PRISE EN COMPTE DE L' ETAT DE LA TORTUE A LA FIN** d'une procédure graphique (ou d'une liste d'instructions) destinée à être intégrée à un projet . Ces trois types constituent les trois modalités du critère ETAT .

Pour une analyse multi-critères d'activités
de programmation en Logo

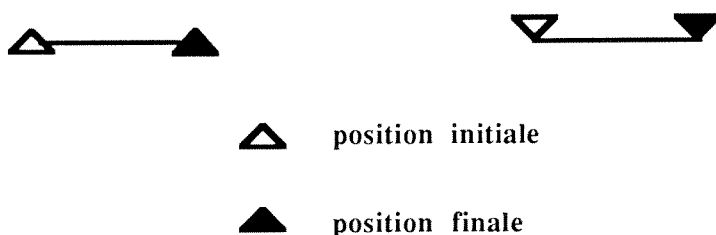
Première modalité du critère ETAT : " PS " .

A la fin de la procédure , il y a un RETOUR A UNE POSITION " STANDARD ". Une position " **standard** " est une position à partir de laquelle la **coordination** de cette procédure avec les autres est **économique** (Il s'agit souvent de la position initiale) . Dans ce cas , il est évident que l'élève a pris en compte l' **état de la tortue** à la fin de la procédure ; la coordination nécessite l'utilisation d' **interfaces** hors de la procédure .

Voici deux exemples de procédures de cette modalité dans la situation du test BALLONS .



Pour réaliser le projet complet , l'élève doit dans ce cas écrire une interface. Les interfaces correspondant respectivement aux deux procédures visualisées ci - dessus peuvent être représentées ainsi :

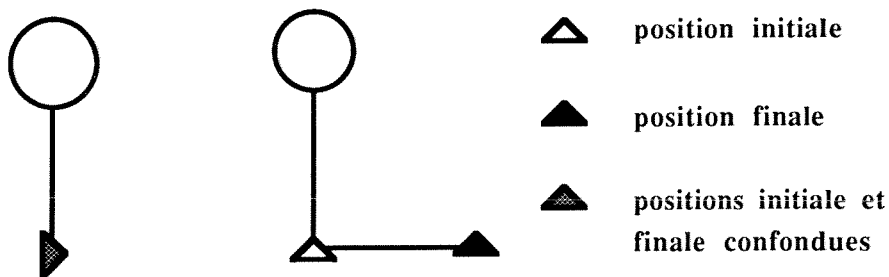


L' utilisation de positions " standard " confère à la procédure un caractère de généralité , d' indépendance du contexte et facilite une éventuelle réutilisation dans d'autres projets .

Pour une analyse multi-critères d'activités
de programmation en Logo

Deuxième modalité du critère ETAT : " INT " .

L'INTERFACE EST INTEGREE à la procédure . Voici deux exemples de procédures de cette modalité dans la situation du test BALLONS .



Dans ce cas aussi , il est évident que l'élève a pris en compte l'état de la tortue à la fin de la procédure .

La réutilisation d'une telle procédure dans un autre contexte peut être difficile .

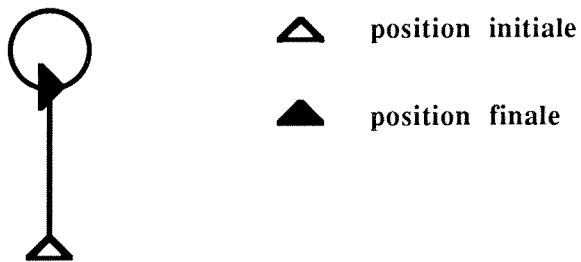
La distinction faite ici entre deux stratégies ne signifie pas pour nous que l'une est la " bonne " et l'autre la " mauvaise " : suivant la situation , l'une ou l'autre peut être la plus économique pour résoudre un problème précis (cf . projet BALLONS) . Mais dans une perspective de travail à long terme et d' **utilisation de procédures hors du contexte** où elles ont été écrites , seul le **retour systématique** à une position " **standard** " est raisonnable .

Troisième modalité du critère ETAT : " SANS " .

AUCUN RETOUR A UNE POSITION " STANDARD " ne figure dans la procédure .
Aucun mouvement de la tortue n'est effectué en plus de ceux qui sont nécessaires au tracé .

Pour une analyse multi-critères d'activités
de programmation en Logo

Voici un exemple de procédures de cette modalité dans la situation du test BALLONS :



4 . 2 Le traitement de la coordination .

Le problème de la coordination des procédures se pose dès qu'il y a décomposition en procédures . La difficulté de la coordination se manifeste ici à deux reprises et de manières fort différentes : d'une part , dans la coordination des procédures résultant de la décomposition que chaque élève a choisie dans ce contexte ; d'autre part , dans la réutilisation , pour tracer les cercles , d'une procédure écrite dans un autre contexte . Cette **différence fondamentale de contexte** nous a conduites à définir deux critères différents :

le critère COORDINATION , qui permet d'analyser le traitement de la coordination entre des procédures de décomposition conçues et écrites dans le contexte où elles sont coordonnées.

le critère COORDINATION HORS CONTEXTE , qui permet d'analyser le traitement de la coordination lors de la réutilisation d'une procédure conçue et écrite dans un autre contexte. Il s'agit ici de la procédure utilisée pour tracer les cercles .

4 . 2 . 1 Le critère COORDINATION .

Les modalités du critère COORDINATION sont :

"0" : la coordination des procédures N'EST PAS REALISEE .

"C " : le programme est CORRECT .

Pour une analyse multi-critères d'activités
de programmation en Logo

Nous distinguerons **deux types d'erreurs** dans la coordination des procédures :

" **ES** " : le programme présente des **erreurs simples** d'interface , comme des erreurs de latéralisation , des erreurs sur la valeur d'un angle ou sur les relations entre des longueurs .

" **EC** " : le programme présente des **erreurs de conception** , comme celles qui résultent d'une conception " image " de la procédure .

Nous désignons par **conception " image "** une identification de la procédure avec le tracé souhaité ou la figure obtenue . L'élève considère la procédure comme un moyen d'obtenir un résultat et non comme une suite d'instructions modifiant (éventuellement) l'état de la tortue . Les premiers programmes des élèves reflètent tous cette conception qui est à rapprocher de leur propre expérience du dessin à la main : pour tracer un trait , il est inutile de savoir où et dans quelle position s'est arrêté le crayon après le tracé précédent .

Cette conception ne peut se repérer uniquement dans l'écriture des procédures . Elle ne peut être observée qu'au moment où l'élève veut coordonner ses procédures . Deux situations sont caractéristiques de cette conception :

- l'élève n'écrit aucune instruction pour coordonner ses procédures alors qu'il n'a pas intégré l'interface aux procédures.
- l'élève écrit une interface supposant un retour à une position "standard" alors qu'il n'a pas fait de retour à une position " standard " .

En principe , le problème de la coordination ne se pose pas si l'élève écrit un programme sans aucune procédure ou instruction , comme REPETE , qui résulte d' une structuration ; mais cette attitude ne s'observe qu'au début de l'apprentissage de la programmation ou pour des programmes très simples . De sorte que , pour ne pas multiplier les codages , nous utiliserons le critère COORDINATION , par extension dans ce cas rare .

4 . 2 . 2 Le critère COORDINATION HORS CONTEXTE (CHC) .

Pour tracer des cercles (approximation par des polygones réguliers) , les élèves disposent de deux procédures écrites dans leurs cahiers:

POLYGONE1 : longueur du coté : nombre de cotés et

POLYGONE2 : longueur du coté : angle .

En effet , nous avons travaillé , lors d'une séance précédente , à l'écriture de procédures permettant de tracer un polygone régulier quelconque , puis un cercle . Tous les élèves ont utilisé l'une ou l'autre des procédures POLYGONE pour tracer les cercles dans la situation BALLONS .

Mais POLYGONE1 et POLYGONE2 sont deux procédures qui ont été conçues, écrites, et exécutées pour répondre à une tâche particulière : le dessin d'un polygone , puis d'un cercle isolé . Les élèves n'avaient pas eu l'occasion de les intégrer dans d'autres projets . Il semble bien que , dans ce cas , la **conception " image "** de la procédure , c'est - à - dire l'identification du cercle avec la procédure POLYGONE , domine chez tous les élèves . Ceci se repère aux faits que :

- Dans la première écriture de leurs programmes BALLON , **aucun élève n'a écrit d'interface** entre les procédures de la tige et du cercle . Leurs cercles étaient donc tous **tangents** à la tige , puisque la position initiale et finale de la tortue est tangente au cercle.



- Au moment de l'exécution , la réaction générale a été la surprise . Les élèves ne s'étaient pas posés la question de la **coordination** à cet endroit alors que tous se préoccupaient de coordination entre **leurs** procédures BALLON .

Pour une analyse multi-critères d'activités
de programmation en Logo

Ces problèmes de **prise en compte de l'état de la tortue** à la fin de la procédure et de **coordination** ne sont donc pas exactement de même nature que ceux qui se posent entre deux procédures qui ont été conçues et écrites pour être coordonnées . La coordination d'une procédure écrite dans un autre contexte peut nécessiter un retour en arrière sur la procédure et un enrichissement de la conception de cette procédure .

C'est pourquoi le problème des cercles est traité à part . Tous les autres critères sont observés **en dehors de la procédure éventuellement utilisée pour tracer les cercles** . Les modalités du critère CHC figurent dans l'annexe 1 .

4 . 3 Un critère pour la répétition .

L'élève peut choisir d'utiliser ou non l'instruction REPETE . S'il l'utilise , il doit déterminer la séquence à répéter et le nombre de répétitions . Mais là encore il a un choix car le traitement de la répétition peut se faire de la manière suivante :

liste d'instructions puis REPETE [la même liste d'instructions] .

Ce type d'écriture de la répétition , que nous avons observé au début de l'apprentissage , est encore présent dans ce test : il persiste donc assez longtemps . Il correspond au **modèle spontané** que les élèves ont de la répétition : **faire une action et recommencer** .

Dans un contexte un peu différent, C. Laborde, N. Balacheff, et B. Mejias (1985) ont aussi observé cette forme de répétition qu'ils rapprochent de " la langue naturelle , dans laquelle la mention de la répétition peut se faire après coup par une expression telle que *continuer ainsi de suite* " .

Les modalités du critère REPETE figurent dans l'annexe 1 .

4 . 4 Deux critères d'utilisation .

SP : Utilisation d'AU MOINS UNE SOUS - PROCEDURE .

" OUI "

" NON "

Pour une analyse multi-critères d'activités
de programmation en Logo

Ce critère ne recoupe pas les autres , puisque l'élève peut choisir d'écrire tout son programme dans une seule procédure , quitte à écrire plusieurs fois une même suite d'instructions . **L' absence de sous - procédure est rarement associée à la réussite d'ensemble d'un projet complexe .**

VAR : Utilisation d'AU MOINS UNE VARIABLE pour désigner une longueur .

" OUI "

" NON "

Si une variable est utilisée pour désigner une longueur , par exemple la longueur de la "tige" , il est possible de **réutiliser** la même liste d'instructions pour les deux sortes de ballons .

5 . MISE EN OEUVRE DES CRITERES .

Les résultats de l'analyse suivant ces critères sont présentés dans le tableau de l'annexe 2.

5 . 1 Pertinence des modalités des critères .

Certaines modalités des critères , possibles a priori dans cette situation , n'ont pas été observées . La modalité SANS retour à une position standard du critère ETAT n'est pas observée , tous les élèves ayant pris en compte la position finale de la tortue suivant l'une ou l'autre des stratégies position standard ou interface intégrée . Il n'y a pas non plus d'erreurs de conception dans la coordination des procédures .

Ce phénomène peut être considéré comme un résultat positif de l'enseignement , car l'absence de retour à une position standard et la conception " image " de la procédure ont été observées chez ces mêmes élèves au cours de l'apprentissage .

Pour une analyse multi-critères d'activités
de programmation en Logo

5 . 2 Profils des individus .

Parmi les huit élèves pour lesquels la **coordination** des procédures est **correcte** , l'instruction REPETE est bien utilisée et les cercles sont bien placés , on trouve **toutes les combinaisons** possibles des critères VARiable et ETAT : certains ont utilisé des variables, d'autres n'en ont pas utilisées ; les uns ont écrit des procédures avec retour à une position " standard " , les autres avec interface intégrée . Mais tous ces élèves ont utilisé au moins une sous - procédure .

Si l'on ne tient pas compte de la position du cercle par rapport à la tige , quatre élèves de plus ont écrit un programme " Correct " . Mais parmi eux , une élève n'a pas utilisé de sous - procédure et un autre n'a pas utilisé l'instruction REPETE .

Si nous croisons les critères **SP : utilisation** d'au moins une sous - procédure et **Coordination** restreint aux modalités Correcte ou non - Correcte , nous obtenons :

Coordination correcte			
		Oui	Non
SP	Oui	11	4
	Non	1	3

La **décomposition** du problème en sous - procédures est **liée à la réussite** d'ensemble puisque sur douze élèves qui ont réussi la coordination , onze d'entre eux ont utilisé des sous - procédures . Cependant la liaison n'est pas statistiquement significative au seuil habituel de 5 % puisque la probabilité , à marges constantes , d'obtenir un élève (ou zéro) ayant réussi sans utiliser de sous - procédure est 0,117 soit près de 12 % .

Deux modalités du critère REPETE a priori possibles (XC et FC) sont absentes mais si l'on considère séparément les deux codes , toutes les modalités sont présentes . L'élève qui n'a pas utilisé l'instruction REPETE , tout en ayant un programme correct , a décomposé en procédures et préfère écrire trois fois les noms de ses procédures .

Pour une analyse multi-critères d'activités
de programmation en Logo

De fait , la situation n'était pas suffisamment complexe pour qu'il soit **absolument nécessaire** , pour réussir , d'utiliser des **variables** , des **sous - procédures** ou même l'instruction REPETE . Pour les variables , il est clair que leur utilisation , qui reste difficile pour les élèves , n'était pas significativement plus économique que l'écriture de procédures sans paramètre .

6 . CONCLUSION .

La grande variété de profils des individus avec seulement dix - neuf élèves et une seule tâche de programmation confirme notre hypothèse de **phénomène multidimensionnel** impliquant une analyse croisant plusieurs critères . Une analyse multi - critères , menée sur plusieurs productions d'un même élève , permet d'enrichir le profil de l'élève en dégagant ses stratégies personnelles de programmation et leur évolution . C'est à cela que nous travaillons actuellement .

La validité de tels critères sera réellement établie si d'autres enseignants ou chercheurs intéressés par ce domaine les mettent en oeuvre dans d'autres situations avec d'autres élèves et en tirent profits et critiques . Nous espérons avoir apporté notre modeste contribution à la grande tâche de la communicabilité des recherches expérimentales .

ANNEXE 1: Modalités des critères CHC et REPETE .

Coordination Hors Contexte :

" 0 " - Aucun cercle n'est tracé .

" C " - Les cercles sont correctement placés aux bouts des tiges : l'interface est correcte .

" T " - Les cercles sont tangents aux tiges : l'interface est absente.

" A " - Les cercles ne sont ni tangents , ni corrects : l'interface est incorrecte.

REPETE un nombre de fois [une liste d'instructions] :

Nous allons définir un critère codé sur deux colonnes , le premier code indique si le nombre de répétitions est exact , le second si la liste d'instructions répétée est correcte ou non .

premier code:

" 0 " : aucune instruction REPETE n'est utilisée .

" C " : l'instruction REPETE est utilisée et le nombre de répétitions est correct .

" F " : l'instruction REPETE est utilisée et le nombre de répétitions est faux .

" X " : l'instruction REPETE est utilisée mais avec une structure du type:
[liste d'instructions] puis REPETE [la même liste d'instructions].

deuxième code :

" 0 " : pas de liste .

" C " : la liste d'instructions répétée est correcte .

" F " : la liste d'instructions répétée est fautive .

Ceci fait que l'on peut obtenir les codes 00 , CC , CF , FC , FF , XC et XF pour ce critère .

Pour une analyse multi-critères d'activités
de programmation en Logo

ANNEXE 2

Les réponses analysées ici sont les programmes modifiés par les élèves après une exécution

CRITERES :	CHC	S P	VAR	ETAT	COor -dination	REpète
ELEVES :						
STé+DAV+KLA	C	OUI	NON	PS	C	CC
SEB	C	OUI	OUI	PS	C	CC
ALI+CAR+NAT	C	OUI	NON	INT	C	CC
MAR	C	OUI	OUI	INT	C	CC
SAR	T	OUI	NON	INT	C	CC
SAB	A	OUI	NON	PS	C	CC
CYR	C	OUI	OUI	PS	C	00
ALE	T	NON	NON	INT	C	CC
FLO	T	NON	NON	INT	ES	CF
GIL	T	NON	NON	INT	ES	XF
KAR	C	OUI	OUI	INT	ES	CF
RAC	T	OUI	NON	PS	ES	CF
REG	T	OUI	OUI	PS	ES	FF
CHR	T	OUI	NON	INT	0	00
MAG	0	NON	NON	INT	ES	CF

Dans ce tableau , nous avons regroupé dans une même ligne les élèves qui correspondaient aux mêmes modalités des critères ; cela ne signifie pas que leurs programmes étaient absolument identiques . Lorsque l'élève ne définit pas formellement de procédure mais utilise un REPETE [une liste d'instructions] , le critère ETAT est observé sur la liste d'instructions .

Pour une analyse multi-critères d'activités
de programmation en Logo

REFERENCES

DUPUIS C. , EGRET M. - A. , GUIN D. (1987) : Logo 3 . Programmation structurée : Présentation et Analyse de situations , Brochure I.R.E.M. de Strasbourg .

HILLEL J. , SAMURÇAY R. (1985) : Analysis of a LOGO environment for learning the concept of procedures with variables , Research supported by Quebec Ministry of Education , FCAC Grant EQ 2539 .

LABORDE C. , BALACHEFF N., MEJIAS B. (1985) : Genèse du concept d'itération : une approche expérimentale , Enfance , Vol. 2 / 3 , pp. 223 - 239 .

MENDELSON P. (1985) : L'analyse psychologique des activités de programmation chez l'enfant de CM 1 et CM 2 , Enfance , Vol. 2 / 3 , pp. 213 - 221.

ROGALSKI J. (1986) : Pour une pédagogie de l'informatique , Enseignement public et informatique , N° 42 , pp. 105 - 109 .