PROGRAMMING RULES: WHAT DO CHILDREN UNDERSTAND?

Celia Hoyles, Richard Noss, Ross Adamson & Sarah Lowe
Institute of Education, University of London, UK

Abstract

We are working to build computational worlds where children can play, design and program videogames. Videogames represent rule-based systems which are expressible in non-textual animated ways. We describe the different ways children (aged 7 to 8 years) articulate a simple rule they have programmed themselves. The results indicate that rule expression is shaped by the type of prompt to make the articulation (to predict, describe or to explain), the narrative context of the game and the medium of expression (computational, spoken or written).

Background

This study concerns children's understandings of rules they have constructed by programming a computer, in the general context of building their own video games. There is much research into children's understanding of causal evidence and logical reasoning, much of it undertaken in the field of developmental psychology. Since the 1970's, evidence has accumulated that children rarely argue solely on the basis of universal laws of logic or domain-independent abstract rules. There are different interpretations as to why this might be the case: that children call up pragmatic reasoning structures derived from experience in context (Cheng and Holyoak, 1985), or that their arguments are derived from knowledge and the way it is structured (Ceci, 1990). How far children are aware of the logical necessity of even their correct conclusions is still a matter of debate.

Following a careful analysis of children's naturally occurring arguments, Anderson, Chinn, Chang, Waggoner, & Yi (1997) report that children tend to omit parts of the logic of their argument (premise, warrant or conclusion), to be cryptic when mentioning the known or obvious, and elliptical when expressing their position, giving no more information than was necessary. Nonetheless, they conclude that the children's arguments were logically complete: that is, the framework used was not inconsistent with deductive logic. The fact that everyday language is likely to be only a partial indicator of reasoning calls into question how children's reasoning might best be investigated. Anderson et al. report that children used words such as 'because', 'so', and 'therefore', in arguments which included *modus ponens*; so simple reference to linguistic pointers is no infallible guide to logical reasoning.

Aside from research on children's logical understanding, we briefly allude to a further strand of research which is an important source for our analysis. This concerns the literature on learning by designing, creating and debugging meaningful 'external' artefacts (the *constructionist* paradigm, Harel and Papert, 1991) and our own work (Noss and Hoyles, 1996). However, an assumption behind our work had been that the expression of complex ideas and its communication to a computer necessitated collections of symbols in the form of textual strings and it was this need for symbolic representation that was crucial for mathematical learning. This assumption is somewhat anachronistic, as we have begun to explore the potential of programming within a non-textual environment in which programs are created by directly manipulating animated characters and animations themselves are the source code of the language.

The study forms part of the *Playground*[1] project (http://www.ioe.ac.uk/playground) in which we are giving children (aged 6 to 8 years) the opportunity to construct creative and fun computer games (see also Harel, 1988 and Kafai, 1995) and at the same time, offer them an appreciation of — and a language for — the rules that underpin them. Broadly, we conjecture that through designing, building and playing their own computer games, children will think differently about ideas like causality and inference, and will find new, more formal and complete ways to express their thoughts about rules. Given the age of the children, it is obvious that we cannot expect them to build symbolic representations of rules. Rather we have based our work on *ToonTalk* (Kahn, 1999) where programming involves the 'training' of animated robots by example, and essentially without text. The conditions which determine subsequent performance of the actions at run time can be generalised or specialised after the training has taken place.[2]

Our research is framed by two questions: i. How is children's understanding and expression of rules mediated by a programming language in which the rules are available for inspection and change and ii. How far do children's descriptions of the rules they have programmed throw light on their understandings of causal reasoning?

The Meaning of a Rule

What do we mean by rules in the context of young children building their own computer games? As long ago as 1932, Piaget was interested in games as complex systems of rules and endeavoured to trace the development of the practice and consciousness of rules in young children (Piaget, 1932). He concluded (largely through an analysis of boys playing marbles), that between ages 7 and 10 years, children were unable to codify rules, although they were able to play games according to social conventions. A conscious realisation of the rules of their games and seeing that they could be changed, Piaget argued, only developed at age 11 –12. For Piaget (and Vygotsky as well), what changes over time is the explicit recognition of the rules of a game.

In our early interviews with children (aged 6 to 8) we asked them to tell us about their favourite games (both on and off the computer) and the rules of these games. We found most only described constraints ("you mustn't hit other children") and not the actual rules designed into the game. From our case studies of the children programming their own games, we have noted that children have indeed become more aware of rules[3], but also that they adhere to different types of rules that we have categorised as follows:

> *Player rules* – a regulation that must not be transgressed: "You must not hit the sides of the maze".

Player goal – a maxim or formula that is generally advisable, but not compulsory to follow: “Score as many points as you can”.

System rules – a generalised statement that describes what is true in most or all cases: “When Theseus touches the minotaur with the sword then he will die”.

Our interest focuses on children’s understanding of the universal system rules that are hard-wired into the computer game (that is, built by programming). Clearly, any programming environment brings its own complexities: e.g. how the rule is expressed in the language, and how virtual ‘phenomena’ like “dying” are translated into programmable actions (like “disappearing”)?

We have engaged in two types of studies to investigate children’s understandings of rules: i. extended case studies tracing the evolution of both games and the children's understandings of the rules (see Goldstein & Pratt, submitted); ii. clinical task-based interviews which probe how children think about rules (in this case, conditions and actions), and the extent to which the expression and application of rules are mediated by the tools available (as well as what light this may throw on existing developmental sequences). The clinical interviews were based around tightly controlled, pre-designed tasks with specific aims (in contrast with the largely exploratory and open computer activities that characterised the case-study work). As well as interacting with the computer, the children, working individually or in pairs, completed worksheets. A researcher, who made metacognitive prompts to probe descriptions, predictions, and expectations, was present with each pair. All the interviews were audio-taped and transcribed. We report on one of the task-based interviews which we conducted with 9 children aged 7/8 years. The interviews took about one hour.

A Task-based Interview: The Cat and the Milk

The aim of the task was to probe children's thinking while programming a simple rule that involved a sequence of condition and action, ‘if touching a particular object then do something’. The first part of the task involved constructing a rule with a warrant between a condition and action that had a common-sense explanation (“if a cat touches the milk it meows”) followed by an application in a less realistic setting.

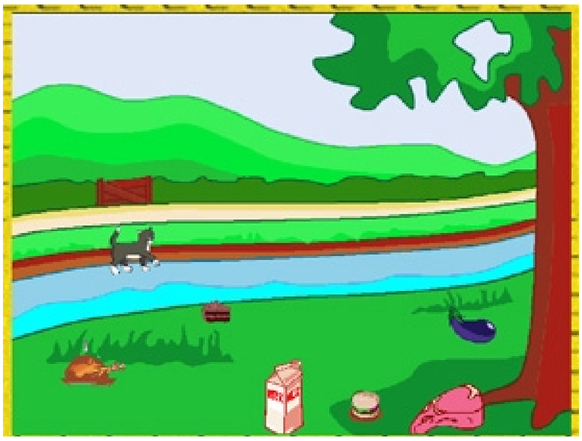

Figure 1: the *cat and milk* task

The children were given the scenario illustrated in Figure 1, in which the cat had already been programmed so that it could be moved about with the mouse. In ToonTalk, this means that a robot had been trained to move with the mouse and placed on the back of the cat. This piece of pre-written ‘code’ we call a *behaviour*: it can be saved on its own but is also transferable to any object, to make it move similarly. The children’s task was to program a robot (to put on the back of the cat) so that if the cat touched the milk carton it would meow. After this, we wanted the ‘move with mouse’ behaviour to be transferred to the milk carton, so it could be moved (a fantasy element) while leaving the meowing behaviour on the cat.

The researcher’s role was to ask the children to describe what they observed on the screen, to explain what was happening and to predict what would happen after programming the new rule. The following prompts were used as guides:

Description/explanation.	Imagine you have to describe what is going on the screen to a friend who cannot see it: What can you see? What is happening? Why is it happening?
Build the rule and describe it.	What can you see? Is it working? What does it do?(so a friend can know)
Add rule and predict what will happen.	What will happen? When will it happen? Why will it happen?
Use rules/behaviours in other contexts	What will happen? When will it happen? Why will it happen?

Specifically, the researcher asked the following questions:

After playing with the cat: “Why can you move the cat about?” *(explanation*)
After training the robot: “Describe what it would do” (*description*)
After putting the behaviour on the back of the cat: “Do you think the cat is going to meow or not when it touches the milk?” (*prediction*)
After trying out the new rule: “What is happening (when they move the cat)? Why?” (*explanation*)
After taking the move behaviour from the cat and giving it to the milk: “What do you think will happen when the milk touches the cat? (*prediction*)
After testing it out: “What is happening? Why?” (*explanation*)

Analysis

After the data was transcribed, we grouped the children's explanations of causality into two general categories. The categories were devised partly from theory (see Donaldson, 1986), but also as a result of a pilot study conducted in 1999.

<u>Category A: Student-centred explanations</u>

A1. Real World (Physical laws): using analogies of everyday events or real events, “it does that as it is a ball and balls bounce.”

A2. Intentional World (Psychological laws): giving the characters motivations and intentions, “cats hate meat so they growl.”

A3. Fantasy (Personal Imagination):

i. shared narrative about the game, “the witch casts a spell on you so you die.”

ii. idiosyncratic.

Category B: Formal computer-centred explanations

These include explanations by reference to the logic of the system or according to the inevitable outcomes from programming. It is possible, however, to distinguish these explanations along a general-specific dimension:

B1. Explanations derived from prior knowledge of general computer systems.

B2. Explanations derived from knowledge of computer games.

B3. Explanations derived from general features of the Playground environment: "there is a behaviour on the back of the bear"

B4. Explanations derived from specific aspects of the programming environment: "the rule is programmed so the robot plays a sound when the bear is touching the cheese".

Results

We present the main types of response to each of the questions and illustrate them with quotes from the children. We also annotate each type of response with a conjecture which is informing our current observations.

Explaining what was observed on the screen: After experimenting with the scene, all the children were asked, "Why do you think the cat is moving?" Four children responded in a way that could be categorised as formal, but general since in ToonTalk all objects are moved by a virtual hand (B3). An example is:

I: Why can you move the cat about?
Joe: Because that's the main thing really.
I: What do you think makes it move?
Joe: Your hand.
I: What makes him move up then?
Joe indicates the mouse.

Another example from Hazel:

I: Why do you think the cat can move?
Hazel: Because you're moving the mouse so it moves the cat as well.
But the fact that neither chose to talk specifically about the behaviours on the back of the cat that made it move, maybe explains their later 'psychological' (A2) as opposed to 'formal' responses, for example:

I: Why can you move the cat about?
Joe: Because it's a creature.
In contrast, the responses of five of the children were categorised as formal/specific; they clearly made the connection between actions on the screen with programmed behaviours.
I: Why do you think the cat is moving?
Sophie: Maybe she already has the behaviour.

However, it was only on inspecting the behaviour in the course of the interview that most children were able to appreciate the mechanism involved. For example, having looked at the back of the cat, Sophie was more specific in her explanation:

Sophie: 'Cos it's got that behaviour 'I move with mouse'.

> Conjecture 1: Children need to link observations with behaviours as a first step to explaining the outcomes of the rules they have observed.

***Describing a new rule*:** After the children had trained a robot to instantiate the rule, 'if touch milk, then meow', we asked them to label the robot so another child would know what it did.

Four children omitted the condition in their descriptions, as illustrated by:

I: First we have to type a label that describes what it does.
Alice: Meows!
I: Jane, is that what you want?
Jane: Meow!
Three children were unsure what to write and wanted to add comments like "this is a fantastic game", while the remaining two wrote complete descriptions. When the behaviour or rule was 'contextualised' by putting it in the back of the cat, *all* the children correctly *described it verbally* e.g:

I: If somebody … found your cat, how would you describe it?
Sophie: It meows when… um… it touches the milk carton…
However, when *writing* a description, their labels varied from uninformative "Fofo" (2 children), to action-only "Meows" (3), to explicit/complete but contextualised, "When the cat touches the milk it meows" (2) to decontextualised and complete, "It meows when it touches the milk." (2). These disparities between what children programmed, what they said and what they wrote, was characteristic of other rule descriptions.

> Conjecture 2: Children are better able to articulate explicitly all the parts of a rule when it is programmed, and/or contextualised in a realistic situation, but the completeness of their expression varies with the medium.

Predicting the consequences of a rule as programmed: We asked the children to predict what would happen when the cat with its new behaviour was moved about the screen. Six children predicted correctly: e.g.

I: Before we play the game, when you move the cat around, what do you think will happen?
Sophie : When it touches the milk carton it's going to make a cat noise.
I: And when it touches the cake, what will it do?
Sophie: Nothing.
I: Try it and see… why is that, by the way?
Sophie: Because we've made a behaviour about the milk carton and not the cake.

Strangely, even after constructing their program and giving it the cat, the others were unable to predict what would happen with any confidence.

I: Now, without you having to press the space bar or do anything, do you think the cat is going to go meow when it touches the milk?
Hazel: No!
I: No, why not?
Hazel: Maybe we have to do a bit more stuff to get it to do it. […]
I: Do you think that cat is going to meow when it touches the milk?
Both: No!
I: Why not?
Joe: Because we haven't ... we've done something wrong.

> Conjecture 3: Even after having constructed a rule explicitly and articulated it in natural language, children may be unsure of its consequences; that is not appreciate its inevitability[4].

Predicting the implied consequences: We then asked the children to put the 'move behaviour' on the back of the milk carton (instead of the cat) and predict if they will hear the meow sound when the milk touches the cat. *All* of the children were either completely certain this would *not* happen, or were very unsure about it:

I: If you move the milk so that it touches the cat, will you hear a meow sound?
Hazel: No, because we haven't told it to do that.
I: We haven't told what to do what?
Hazel: We haven't told the milk . . when it touches the cat, say the milk runs around, we haven't told it that when it catches the milk [cat? CH] *it says meow.*
I: We haven't told the milk that when it touches . . .
Hazel: We haven't flipped the milk over and when, so we haven't wrote when you touch the cat it will say meow.

It seems that the implications of the rule (that is, the symmetry of the formal relation of touching) did not 'transfer' to an unfamiliar situation of milk moving.

> Conjecture 4: Programming the rule correctly and seeing its consequences, does not necessarily lead to a formal understanding of all the rule's implications.

Explaining what was observed on the screen: After seeing what actually happened and hearing the meow when the milk was moved to the cat, all the children began to appreciate the symmetrical nature of touching, albeit often with an element of psychological causality:

Jane: I know why. 'cos the cat is touching the milk, and the milk is touching the cat.
I: When you move that milk to the cat, what happens?
Jane: Meowing.
I: Now, who's meowing?
Jane: The cat.
I: Why does the cat meow?
Jane: Because he likes the milk!

Conjecture 5: Playing with the rule and observing its implications, begins the process of appreciating the logic of the implied consequences of a new rule.

Conclusions

Overall when the mechanism of the behaviour is articulated and 'connected' to an object, children are better able to explain and predict what is happening in their game and come up with a rich mix of formal, narrative and psychological explanations. Yet even if children program a rule correctly, they may not express it completely in spoken or written language, especially in a decontextualised form. More interestingly, children may not predict the implied consequences of a formal rule they have programmed, particularly when the narrative does not 'make sense'. This suggests that the formal means of expression may not yet be fully integrated with verbal and written articulations. We have noticed that with more experience, children are able to integrate their various descriptions and to make simultaneous and multiple interpretations of what is going on – in terms of the narrative requirements of the game (the cat needs the milk to be healthy) and the formal requirements of the rules they program into the game.

References

Anderson, Richard C, Chinn Clark, Chang Janice, Waggoner Martha, Yi Hwajin (1997). *On the Logical Integrity of Children's Arguments*. Cognition and Instruction, 15(2), pp. 135-167.

Ceci, Stephen J (1990). *On Intelligence…more or less. A bio-ecological Treatise of Intellectual Development.* Century Psychology Series, Prentice Hall, USA, ISBN 0-13-634205

Cheng, P.W., & Holyoak, K.J. (1985). *Pragmatic Reasoning Schemas.* Cognitive Psychology*, 17, 391-416.*

Donaldson, Morag (1986) *Children's Explanations* Cambridge: Cambridge University Press.

Goldstein, R. and Pratt D. (submitted to PME 25). Michael's Computer Game: a Case of Open Modelling.

Harel I. and Papert, (1991) *Constructionism.* Norwood: Ablex

Harel, I. (1988). *Software Design for Learning: Children's Constructions of Meanings for Fractions and Logo Programming.* Unpublished Doctoral Dissertation. Cambridge MA: MIT Laboratory.

Kafai, Y.B. (1995). *Minds in Play: Computer Game Design as a Context for Children's Learning*. Erlbaum.

Kahn, K. (1999). *Helping Children Learn Hard Things: Computer Programming With Familiar Objects and Activities*. In Druin, A. (ed) The Design of Children's Technology. San Francisco: M. Kaufman 223-241.

Noss, R., & Hoyles, C. (1996). *Windows on Mathematical Meanings: Learning Culture and Computers*. Dordrecht: Kluwer.

Piaget, Jean. (1932) *The Moral Judgment of the Child.* International Library of Psychology Philosophy and Scientific Method, Kegan Paul, Trench, Trubner & Co. Ltd., London.

---

[1] European Union *Esprit* Grant No: 29329.

[2] Details of how ToonTalk works, its design principles and some applications can be found at http://www.toontalk.com/

[3] We should note here that there are many facets to a game other than rules, which are just as important in game construction and play, namely constraints, descriptive and declarative aspects.

[4] We recognise here that there may well be an element here of lacking confidence in their programming skills.