Algebraic Equations as a Creative Medium

Ken Kahn[1] and Niall Winters[2]

Abstract

We describe MoPiX, a software construction kit for making graphics, animations, simulations, and games using equations. Children using MoPiX relate to equations as a means of expressing themselves creatively. Equations empower MoPiX users to make interacting objects move, spin, and change size, colour and shape. Objects can leave trails as they move. Interactive applications and games can be created containing objects whose behaviour is a function of the state of the mouse or keyboard. MoPiX can be used for "serious" purposes such as implementing (and learning about) Newtonian mechanics or for playful creations of colourful animated works of art. Collaborative creations are well supported due to the extreme modularity of applications built upon algebraic equations.

Keywords Software construction kits, equational programming, MoPiX

1. INTRODUCTION

Children have been creating graphics, animations, simulations, and games using child-friendly programming systems since the late 1960s. The work reported here relies upon equations, rather than computer programs, to describe dynamic behaviours. Alternatively, MoPiX can be described as using algebra as a programming language.

For many children algebra is experienced as boring and confusing manipulations of symbols that have no personal connection with their lives and aspirations. By using MoPiX, equations become empowering. Children come to see equations as a way to express animations and games. Furthermore, children can *use* equations before acquiring an understanding of how to create or manipulate them. A young child's first experiences with equations may be as "magical incantations" that cause things to move, change their appearance, or draw a trail. Given a rich library of pre-defined equations, they can use MoPiX to create a wide variety of animations or simulations without first mastering algebra. A child who delves deeper into the *meaning* of equations will acquire greater expressive power by customizing equations and creating new ones.

2. EQUATIONS IN MOPIX

MoPiX is built around the surprisingly expressive equations of the form

$$f(object_i,t) = \ldots$$

[1] ToonTalk@gmail.com, University of Oxford, Oxford, UK.
[2] n.winters@ioe.ac.uk, London Knowldege Lab, London, UK.

These equations always define functions of two arguments: an object and the time. For example,

$$x(object_1,t) = t$$

states that $object_1$ moves to the right at a constant speed. And

$$y(object_1,t) = t \cdot 2$$

states that it should also move upward at double speed. Combining the two equations causes the object to move upwards at a 30° angle.

Equations can be difference equations that incorporate the previous state or the state of other objects (e.g. for a pen to plot changes to another object).

MoPiX consists of the following components:

- An engine that computes the value of functions for any object at any time using the equations the user has associated with objects.
- An animation engine that, for any integer t, displays all the objects with the state of their display attributes at time t. There are display attributes for the position, orientation, size, red, green, and blue colour components, transparency, and shape. Furthermore, there are pen attributes that specify attributes of a trail, if any, an object leaves as it moves. Typically the real-time elapsed between t and t-1 is small enough that a high frame rate is achieved resulting in smooth animation.
- A work area where users add and remove equations to and from objects
- An equation editor
- A library of equations

3. CONCLUSIONS

Our initial experiences confirm that, indeed, algebraic equations can be used as an expressive medium to create animations, simulations, and games. Programs created include Newtonian mechanics, graphing, abstract animation, and a juggling game. Equations can be combined in creative and surprising ways to achieve ends that matter to children. Perhaps they will acquire a positive attitude towards algebra after experiencing equations as empowering and maybe learn algebra while having fun.

4. AVAILABILITY OF THE SOFTWARE

MoPiX is freely available at http://mopix2.appspot.com. It is implemented in Java and uses GWT (http://www.gwtproject.org/) to translate the client code to JavaScript in order to run in any browser. Source code is under an open source license and is available at https://code.google.com/p/modelling4all/source/browse/#svn%2Ftrunk%2FMoPiX.

The full version of this short paper can be found at https://drive.google.com/file/d/0B0taMM6vlEqQbDJSSzFnYk1ZRTA/edit?usp=sharing or http://tinyurl.com/mopix-paper.