

Meeting Minutes 31 July 2001

More on Equation Scope and Initial Conditions

Author:

Warren Hedley (Bioengineering Institute, University of Auckland)

Contributors:

David Bullivant (Bioengineering Institute, University of Auckland)

David Nickerson (Bioengineering Institute, University of Auckland)

Poul Nielsen (Bioengineering Institute, University of Auckland)

1 Introduction

A meeting was called on 31 July 2001 to discuss some particularly nasty mathematics that Yi had given us. The problem revolved around a model that very much depended on a sequential system of operations.

We had a meeting this afternoon to go over the problem that you sent me this morning (quoted at the bottom of this message, if anyone hasn't seen it) and came to the following conclusions.

The model you sent us essentially is computer code, and relies on a sequence of operations to be solved reliably. If you want a model like this to perform as advertised then it should be distributed as computer code, not using CellML. CellML, by its use of MathML, does not allow the specification of a sequence of operations, but only a system of equations, some of which might be qualified with some conditions (such as initial conditions).

There is also a serious problem in the fourth equation in your demo, reproduced below:

```
mass = (0.5 < flag && flag < 1.5) ? (mass/2.0) : mass;
```

In this particular equation, the equals sign actually represents an assignment operation, not an equality which is what we generally mean. You should really right this as follows:

```
mass <- (0.5 < flag && flag < 1.5) ? (mass/2.0) : mass;
```

MathML does not contain an assignment operator, and to specify the above equation using an `<eq />` element would represent a serious mathematical invalidity.

We still considering what is the actual underlying mathematical model that drives the example you gave us, but haven't made much progress yet. If you have any ideas, they'd be much appreciated.