C-32: Computational implementations of hierarchical structures in building design and structural engineering

W P S Dias
(Dept of Civil Engineering, University of Moratuwa)

Virtually all real world entities are arranged in hierarchical structures. Some of these hierarchies are specialisations, and most members of such hierarchies are classes, as opposed to specific instances. One of the key features of specialisation hierarchies is attribute inheritance. This property of attribute inheritance has been implemented in the object oriented computational paradigm. Object-oriented databases are very efficient for storing data corresponding to specialisation hierarchies. A practical example of this, described in the paper, is the creation of entity libraries for product models.

The other type of hierarchy we observe in the world is the decomposition hierarchy. One of the key properties of such hierarchies is the emergence of new properties as one goes up the hierarchy (i.e. in the aggregation as opposed to the decomposition direction). Computer implementation of emergence, is more difficult. At a very basic level, the finite element method can be said to implement emergence, in that properties of the whole structure are built up from those of component elements; the quality of emergence is however not very rich.

Another example of emergence used in structural design is the use of generative grammars to generate layout diagrams; here, primitive geometric elements and combination rules result in shapes that can have qualitatively higher semantic

contents. Finally, the genetic algorithm (GA) paradigm, used both for structural optimisation and for structural design, also embodies the concept of emergence in that behaviour at the phenotype-level emerges from component elements defined at the genotype level. The paper presents some of these GA exercises.