Chapter 8.5
Adaptive Computation Paradigm in Knowledge Representation: Traditional and Emerging Applications

Marina L. Gavrilova
University of Calgary, Canada

ABSTRACT

The constant demand for complex applications, the ever increasing complexity and size of software systems, and the inherently complicated nature of the information drive the needs for developing radically new approaches for information representation. This drive is leading to creation of new and exciting interdisciplinary fields that investigate convergence of software science and intelligence science, as well as computational sciences and their applications. This survey article discusses the new paradigm of the algorithmic models of intelligence, based on the adaptive hierarchical model of computation, and presents the algorithms and applications utilizing this paradigm in data-intensive, collaborative environment. Examples from the various areas include references to adaptive paradigm in biometric technologies, evolutionary computing, swarm intelligence, robotics, networks, e-learning, knowledge representation and information system design. Special topics related to adaptive models design and geometric computing are also included in the survey.

INTRODUCTION

Adaptive computing focuses on the methodology and implementation of algorithms and systems that can adjust to different situations and circumstances. An adaptive system may change its own behavior depending on the goals, tasks, and other features of individual users and the environment. Adaptivity is important for ubiquitous and pervasive computing, and as it will be shown in this survey, plays an important role in a variety of traditional as well as emerging areas, such as biometric technologies, evolutionary computing, swarm intelligence, robotics, networks, e-learning, knowledge representation and information system design.

The constant demand for complex applications, the ever increasing complexity and size of software systems, and the inherently complicated nature of the information drive the needs for developing radically new approaches for information representation and processing. This drive is leading to creation of new and exciting interdisciplinary fields that investigate convergence of software science and intelligence science, as well as computational sciences and their applications. As can be seen from the definition, the driving force behind the need for adaptive paradigm is variety of situations, variability in backgrounds and needs of different user groups or applications. This survey article presents the new paradigm of the algorithmic models of intelligence, based on the adaptive hierarchical model of computation, and presents the algorithms and applications utilizing this paradigm in data-intensive, collaborative environment.

ADAPTIVE METHODS IN TERRAIN MODELING

For a long time, researchers were pressed with questions on how to model real-world objects realistically, while at the same time preserving efficiency, quality and operability requirements. The examples from the area of computer graphics and terrain modeling showcase the concept perfectly. Over the past twenty years, a grid, mesh, TIN, k-d trees, and Voronoi based methods for model representation were developed (Bonnefoi and Plemenos 2000, Gold and Dakowicz 2006, Cohen-Or and Levanoni 1996, Duchaineauy et. al. 1997, Franc and Skala 2002, Iglesis 2002, Kolingerová 2002). Most of these were however static methods, not suitable for rendering dynamic scenes or preserving higher level of details (see Figure 1.). In 1997, first methods for dynamic model representation: Real-time Optimally Adapting Mesh (ROAM) and Progressive Mesh (PM), were developed (Duchaineauy 1997). However, even with the further improvements (Li et. a. 2003), these methods were not capable of dealing with large amount of complex data or significantly varied level of details (see Figure 2.). The main difference between terrain visualized using static and adaptive methods is the size and distribution of the triangles – in Figure 1, it is clearly seen that the patches of similar triangles are used throughout the various terrain features, while Figure 2 uses adaptive methods to decide on the most appropriate triangle sizes based on the curvature and distance from the viewer. However, this method is still not sufficient for dealing with all variety of terrain features, nor it is fast enough to be used in real-time.

Recently, the adaptive multi-resolution technique for real-time terrain rendering was developed (Apu and Gavrilova 2005). The method is characterized by the efficient representation of massive underlying terrain, utilizes efficient transition between detail levels, and achieves frame rate constancy ensuring visual continuity. The method is based on the adaptive loop subdivision and recursive split operation (see Figure 3.), implemented with the use of novel S-Queue operations ordering data structure.

Furthermore, a novel approach based on adaptive dynamic viewer-dependent level of details (LOD), utilizing the above strategy, was developed for real-time terrain rendering. The approach uses mesh regularity operator and LOD control parameters to achieve fast recursive seamless patch stitching, ensure geometric regularity, improve rendering quality, provide multi-resolution storage and allow for rendering and transmission of massive data sets (see Figure 4).

More formally, the process can be described as follows. A mesh M can be viewed as a piecewise linear surface. It is defined as a pair (K, V) where $V \subset \mathbb{R}^3$ is the set of vertices and K is a simplicial complex specifying the connectivity of the mesh simplices (the adjacency of the vertices, edges and faces). A combinatorial k-simplex of K is the $(k + 1)$ element subset of K. Therefore the 0-simplices $\{i\} \in K$ are called vertices, the 1-simpices $\{i, j\} \in K$ are called edges, and the 2-simplices $\{i, j, l\} \in K$ are called faces.