Inventor

Jason William Staiert
Des Moines, Iowa
United States of America

Title of Invention

Method for simplifying hexagon and square terrain map pathfinding, adjacency, and presentation algorithms when edge (and vertex) properties are important.

Background of Invention

This invention pertains to the representation, processing, and presentation of terrain in computer simulations and games using a hexagon or square grid of adjacent cells where edge (and vertex, square maps only) data is important to pathfinding, adjacency calculations, and presentation of the terrain to the user.

A state-of-the-art implementation stores edge and vertex data in structures separate from that used to store the properties of each hex or square in the map (see Figure 1). Pathfinding and adjacency algorithms must reference these separate data structures using specialized code that increases the complexity of the algorithm and potentially results in more cache misses due to accessing memory outside the cache working set. Presentation algorithms are subject to the same problems with the added complication of requiring computer generated geometry to represent edge and vertex data, and to stitch adjacent cells together, before rendering (see Figure 2).

Furthermore, some applications require edges and vertices to be traversable. In the state-of-the-art, this is handled in a number of ways, always requiring specialized procedures and possibly specialized data structures.

Summary of Invention

This invention eliminates the need for specialized code for pathfinding and adjacency calculation, and increases the likelihood of the data being in-cache, by integrating edge (and vertex, square maps only) data as specialized cells within the hexagon or square map. Two and three dimensional presentation algorithms are similarly improved with the added benefit that computer generated geometry, required to present edge and vertex data, and to stitch together adjacent cells, before rendering is substantially simplified in the worst case or no longer required in the best case. Furthermore, this invention eliminates the need for specialized procedures and data structures for traversing edges and vertices.

Summary of Drawings

Figure 1 - state-of-the-art storage of map data in a computer system

Figure 2 - state-of-the-art presentation of map data

Figure 3 - relationships of original map cells to new in-map representation

Figure 4 - transfer of cell data from original map to in-map representation

Figure 5 - transfer of edge data from original map to in-map representation

Figure 6 - transfer of vertex data from original map to in-map representation

Figure 7 - smooth blending between hexagon in-map representation of edge geometry

Detailed Description

It is assumed that the user of this invention begins with a conventional hexagon or square terrain map, with edge and/or vertex data (as shown in Figures 1 and 2) and encoded in some digital or physical form.

This invention defines a conversion process that converts a conventional hexagon or square map with edge (and vertex, square map only) data to a conventional hexagon or square map without edge and vertex data, and twice the resolution of the original map. This new map is what is meant by an “in-map” representation. The steps of this process are:

1) A blank conventional hexagon or square map, with twice the resolution of the original map, is generated as shown in Figure 3. This new map does not contain edge or vertex data.
2) All cell data of the original map is transferred to the in-map representation as shown in Figure 4.
3) All edge data of the original map is transferred to the in-map representation as shown in Figure 5.
4) For square maps only, all vertex data for the original map is transferred to the in-map representation as shown in Figure 6.

The in-map representation classifies cells as type one of; regular, edge, or vertex. Further sub-types are specified based on the needs of the specific application. For example, a sub-type for an edge cell may be “river” which is further sub-typed by “direction-of-flow".

This invention defines an adjacency determination algorithm that operates on the in-map representation and takes into account the type of cell (regular, edge, or vertex) used as the source, and a mode of traversal based on cell type and zero or more sub-types.

This invention defines a pathfinding algorithm that operates on the in-map representation and utilizes the aforementioned adjacency determination algorithm and is configured by cell type and/or zero or more sub-types.

This intention defines a presentation algorithm that operates on the in-map representation. It selects and customizes the geometry for each cell based with limited or no dependencies on the geometry of adjacent cells.

This invention defines an extension to the aforementioned presentation algorithm that allows smooth blending of hexagon map edge cells at the vertex location of the original map as shown in Figure 7.