CSCI 540 - Advanced Databases Course Project Progress Report for November 22nd

Larry Lynn
Liessman Sturlaugson
Cole Schock

November 22, 2011

Abstract

During the preceding week, the iMinMax team integrated several different components together for use with their iMinMax implementation, including the command line parsing library, the CSV parser, and the STX B$^+$-Tree library. With these components in place, the main program under development is able to load a CSV file passed from the command line, populate the points data structure, insert data points into the B$^+$-Tree, save the tree to disk, and load the tree from disk.

1 Command Line Parser Integration

First, the command line parsing library [2] was integrated into the main() method and the switching structure to support the required command line flags was set up.

Currently, the only option supported is reading in a given CSV file. Because the program must be able to be scripted, the team also wrote a test script to iterate through multiple CSV files. Other test scripts will also be written as the main program progresses.

2 CSV Parser Integration

Second, the CSV parsing code (as adapted from [3]) was added to an iMinMax utility class for populating the data structure indexed by the B$^+$-Tree values. As per the class discussion last week, the iMinMax team's code maintains a data structure for holding all of the point data and uses the B$^+$-Tree to hold just the indices of the points in the data structure.

The iMinMax team also intends to use the CSV parser and data structure to easily separate the tree creation time from the index creation time, as required by the project performance measures.

3 STX B$^+$-Tree Integration

Third, the STX B$^+$-Tree library [1] was incorporated into the iMinMax class and the load/insert/save functionality was tested.

The btree.h file was augmented with the code from Tim Wylie for counting the nodes accessed during a query. The native tree statistics were also tested for giving the total number of nodes and number of levels in the tree.

4 C++ IMinMax Class

The class holding iMinMax functionality was also begun. The team proposes to use this class to hold and manage access to the point data structure and the B$^+$-Tree, as well as hold the methods for querying the tree, saving and loading the tree, and collecting tree statistics.

5 Median Approximation

Finally, the iMinMax team has also begun discussing approaches for automatically calculating θ and eventually θ_i for each dimension. However, the team has decided to focus on getting the core requirements in place before adding this extension. Nevertheless the team is aiming to write the code such that the extension allowing multiple θ_i can be added without having to rewrite too much existing code.

References

[1] Timo Bingmann. STX B+ Tree C++ Template Classes, November 2011.

[2] Daniel Gutson and Hugo Arregui. getoptpp - Yet Another getopt C++ version, STL-streaming like, November 2011.

[3] Jonathan Leffler. How can I read and manipulate CSV file data in C++?, November 2011.