Empirical Score of The Static Competitive Ratio For “Self-Organizing Binary Search Trees”

This work concerns calculating the “empirical score” of certain theorems, consisting of the comparisons between the four algorithms:

1. **Static Optimal Binary Search Tree algorithm**
   (***Introduction to Algorithms 3rd Edition*** *- Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein)*
2. **Splay Tree Algorithm**
   (A survey of self-organizing data structures - Albers, Susanne and Westbrook, Jeffery)
3. **Move to Root**
   (A survey of self-organizing data structures - Albers, Susanne and Westbrook, Jeffery)
4. **Dynamic Monotone**
   (A survey of self-organizing data structures - Albers, Susanne and Westbrook, Jeffery)

Starting each time with a randomly-built BST on the distinct keys (1, …., n}, n=1000. This is done using standard insert of keys into an initially empty tree. Access sequences of keys will be non-uniformly biased as follows:

> Generate n integers a_i where each is a random integer from 1 to 100, and with c_i being the sum of the first ‘i’ generated integers, c_0 = 0 and A = c_n. Obtaining a random integer j from 1 to A and “access” key k if c_(k-1) < j <= c_k.
>
> **Note:** all accesses will be successful searches

Access sequences will be 10,000 long and while keeping track of the number of accesses to each key, so that once it ends, a static optimal BST can be created with those frequencies instead of probabilities. Costs are defined based on the survey paper **(A survey of self-organizing data structures -** Albers, Susanne and Westbrook, Jeffery).

Computing the empirical score of the static competitive ratio, in particular the average over 10 runs. This is done for both of the memoryless self-organizing BSTs, and compared to the theorems in the paper while making observations on the empirically observed static competitive ratio of the state-based algorithm relative to the memoryless algorithms.

.