Samuel Dickerson

17 April 2024

Dr. Spickler

COSC 320-001

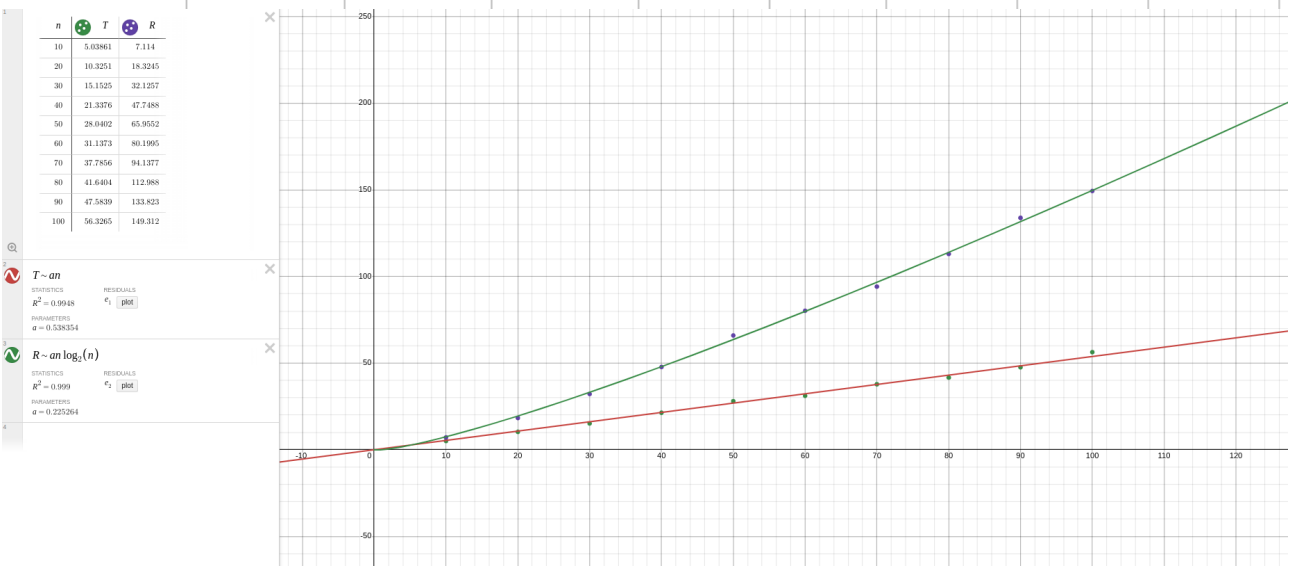
# The Effect of Tree Size, Element Range, and Number of Insertion and Deletion Cycles on Time to Insert and IPL of AVL and Red-Black Trees

The three characteristics of the program that were tested had intriguing correlations. When tree size was the variable being changed, and element range and cycles stayed constant, the time to insert changed dramatically, and the cycle time and IPL time changed slightly for only the Red-Black tree. This is because the AVL tree kept the same number of nodes regardless of changing the tree size. However, for inserting the AVL still took more time with more nodes, because each node calls a balance on the nodes above it. Next, when element range changed but tree size and amount of insertion and deletion cycles remained the same, the output that remained consistent was the average IPL of the Red-Black tree, the rest changed, and will be explained further later. Lastly, when the amount of insertion and deletion cycles was the variable being changed, the time to insert remained consistent (as expected because insertion is timed prior), the cycle time increased, also as expected, and the path length did not change. Altering insertion and deletion cycle amount had no overall impact on the other facets of the testing.

When testing the impact of tree size, the size started at 10 million and went up to 100 million, and the element range and number of cycles was 1 million. The insertion of the AVL tree increased by around 5-6 seconds per 10 million nodes, and demonstrated a linear relationship. The RB tree demonstrated a steeper curve than linear, and appeared to be an*lg(n). The time for one million cycles of the AVL tree did not change in any significant manner as a result of the tree size, however the Red-Black tree did increase from 1.99 seconds at 10 million nodes to 3.77 seconds at 100 million nodes. This result was interesting because from the code it could be inferred this relationship would be the opposite for the Red-Black tree. The while loop in the insertion and

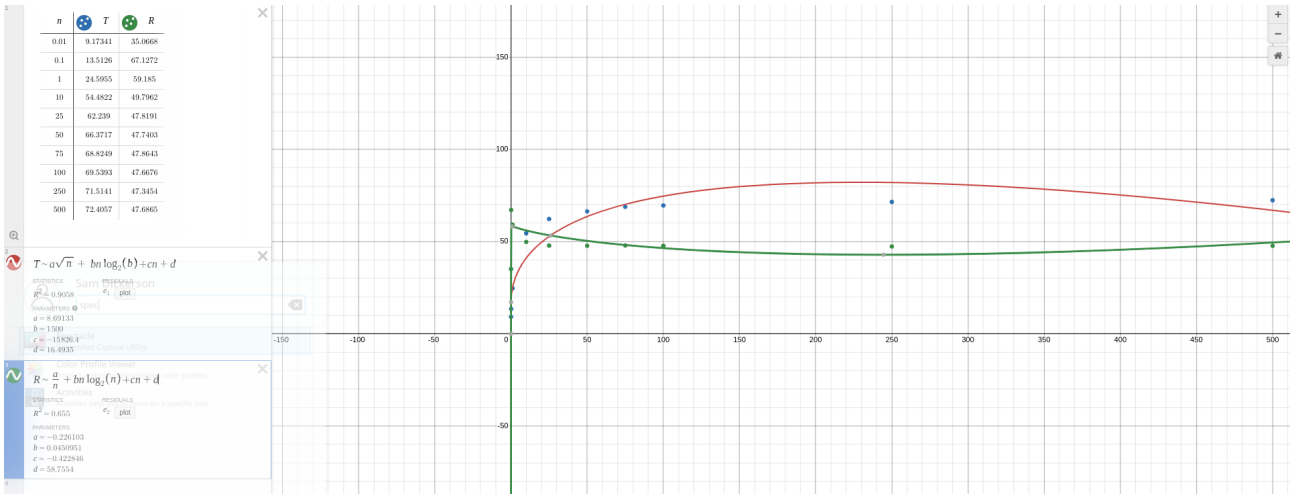
deletion cycles would run more when there are less nodes in the tree because it would not find the node to be deleted. However, since the element range was already smaller than the smallest tested tree size, there was no shortage of nodes to delete and the node was probably found almost every time, so another factor probably increased the Red-Black insertion/deletion cycle time. The average IPL also increased slightly in the same manner as the amount of cycles. The average started at 23.8 at 10 million nodes, but rose to 27.3 at 100 million nodes. This makes more sense than the insertion/deletion cycles, because more nodes leads to a larger tree which leads to a longer height.

| Tree Size (millions) | Element Range | Cycles (million | AVL Insert (s) | RB Insert (s) | AVL Cycle | RB Cycle (s) | AVL IPL | RB IPL |
|---|---|---|---|---|---|---|---|---|
| 10 | 1 | 1 | 5.03861 | 7.114 | 1.10183 | 1.99189 | 19.3945 | 23.8081 |
| 20 | 1 | 1 | 10.3251 | 18.3245 | 1.09729 | 2.37876 | 19.4146 | 24.8907 |
| 30 | 1 | 1 | 15.1525 | 32.1257 | 1.15665 | 2.65847 | 19.4094 | 25.5237 |
| 40 | 1 | 1 | 21.3376 | 47.7488 | 1.08469 | 2.79784 | 19.4072 | 25.9633 |
| 50 | 1 | 1 | 28.0402 | 65.9552 | 1.25187 | 3.17164 | 19.4416 | 26.3022 |
| 60 | 1 | 1 | 31.1373 | 80.1995 | 1.21784 | 3.19596 | 19.4428 | 26.6482 |
| 70 | 1 | 1 | 37.7856 | 94.1377 | 1.18093 | 3.13779 | 19.443 | 26.8312 |
| 80 | 1 | 1 | 41.6404 | 112.988 | 1.09468 | 3.55469 | 19.4209 | 27.0036 |
| 90 | 1 | 1 | 47.5839 | 133.823 | 1.21362 | 3.28676 | 19.42 | 27.1394 |
| 100 | 1 | 1 | 56.3265 | 149.312 | 1.20664 | 3.77492 | 19.4302 | 27.3256 |

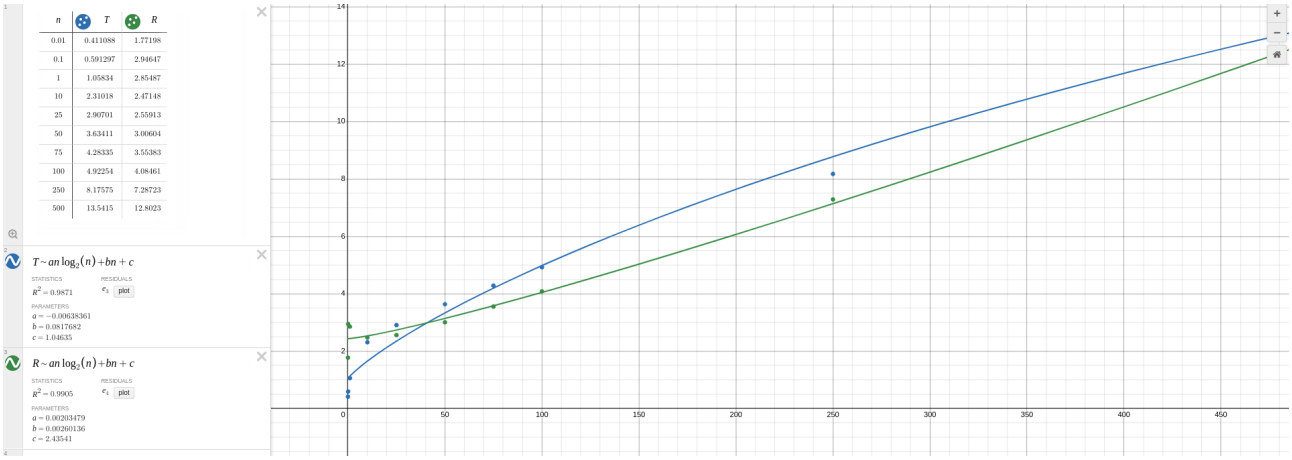



The change in element size resulted in the most data values being changed. When testing element range, the time to insert was affected for both the AVL and Red-Black Tree. Both insert curves were difficult to fit. The AVL times jumped far at first, and then leveled out as the node times got increasingly higher. The jump from 1 million nodes to 10 million nodes was around 30 seconds, while the jump from 10 million to 100 million was only 15 seconds. There did not seem to be a regulated pattern to the insert. The Red-Black tree was also interesting. The first element range tested was 10,000, and the Red-Black tree was able to insert the nodes in 35.06 seconds. When the

element range was increased to 100,000, the tree took 67 seconds to insert. From there, any increase decreased the time, until the element range reached 25 million nodes. From 25 million nodes to 500 million nodes, the red black tree leveled off and took 47 seconds for each test. This spike at the beginning and then leveling off at the end was difficult to find any kind of curve fit for.



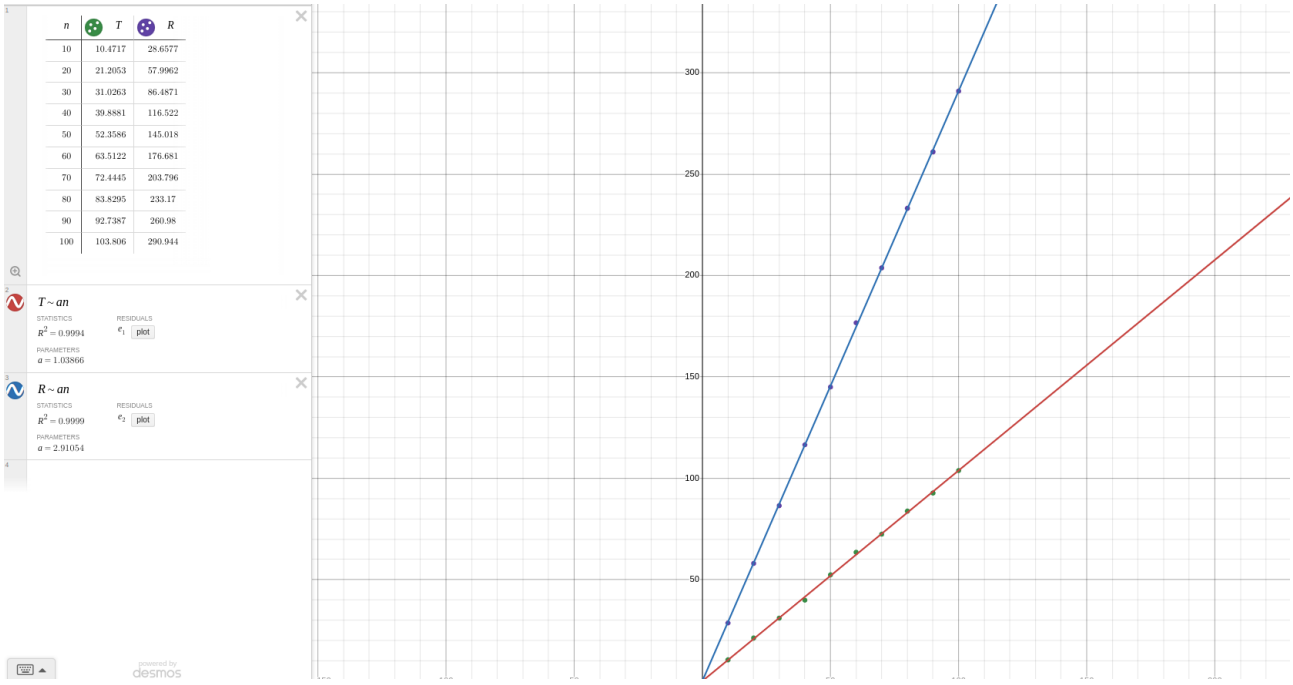

The AVL cycle time and RB cycle time was affected by the element size, unlike tree size. The cycle time did not change dramatically—even at 500,000 million nodes it was still only 13 seconds for the slowest time for either tree. The cycle time increased as the element range increased, so it can be concluded that there were many attempts to delete a node not in the tree, and this increased as the element range increased. Since the AVL tree is only as large as the element size, this caused the cycle time to increase. A good fit for both curves was an*lg(n) + b*n + c.




The average IPL increased for the AVL tree as mentioned before, however, the IPL for the Red-Black tree did not change. This is because the Red-Black tree IPL is determined by tree size, not element range.

| Tree Size (millions) | Element Range | Cycles (s) | AVL Insert (s) | RB Insert (s) | AVL Cycle | RB Cycle | AVL IPL | RB IPL |
|---|---|---|---|---|---|---|---|---|
| 50 | 0.01 | 1 | 9.17341 | 35.0668 | 0.411088 | 1.77198 | 12.6587 | 26.3937 |
| 50 | 0.1 | 1 | 13.5126 | 67.1272 | 0.591297 | 2.94647 | 16.0303 | 26.3794 |
| 50 | 1 | 1 | 24.5955 | 59.185 | 1.05834 | 2.85487 | 19.4231 | 26.2936 |
| 50 | 10 | 1 | 54.4822 | 49.7962 | 2.31018 | 2.47148 | 22.8332 | 26.1827 |
| 50 | 25 | 1 | 62.239 | 47.8191 | 2.90701 | 2.55913 | 23.9368 | 26.1311 |
| 50 | 50 | 1 | 66.3717 | 47.7403 | 3.63411 | 3.00604 | 24.4857 | 26.1041 |
| 50 | 75 | 1 | 68.8249 | 47.8643 | 4.28335 | 3.55383 | 24.6748 | 26.1239 |
| 50 | 100 | 1 | 69.5393 | 47.6676 | 4.92254 | 4.08461 | 24.8081 | 26.0858 |
| 50 | 250 | 1 | 71.5141 | 47.3454 | 8.17575 | 7.28723 | 25.0414 | 26.1326 |
| 50 | 500 | 1 | 72.4057 | 47.6865 | 13.5415 | 12.8023 | 25.1265 | 26.1197 |

Finally, when the amount of cycles was the variable being changed, all data remained the same except for the AVL and RB cycle time. Changing the amount of cycles was the least significant out of the three variables tested. It was straightforward, similarly to tree size, the AVL cycle time rose steadily as the amount of cycles increased, but unlike the tree size testing, the RB cycle time also rose linearly. The process of re-balancing a Red-Black tree on insertion and deletion is more complicated than an AVL tree, so it follows that it would take longer per cycle for the Red-Black tree.




| Tree Size (millions) | Element Range | Cycles (million | AVL Insert (s) | RB Insert (s) | AVL Cycle | RB Cycle | AVL IPL | RB IPL |
|---|---|---|---|---|---|---|---|---|
| 50 | 1 | 10 | 24.5846 | 59.2442 | 10.4717 | 28.6577 | 19.4123 | 26.2899 |
| 50 | 1 | 20 | 24.791 | 59.2127 | 21.2053 | 57.9962 | 19.4246 | 26.3118 |
| 50 | 1 | 30 | 24.3237 | 59.155 | 31.0263 | 86.4871 | 19.409 | 26.267 |
| 50 | 1 | 40 | 24.9191 | 59.7368 | 39.8881 | 116.522 | 19.4745 | 26.3234 |
| 50 | 1 | 50 | 24.7003 | 59.1445 | 52.3586 | 145.018 | 19.4035 | 26.2774 |
| 50 | 1 | 60 | 24.959 | 59.8021 | 63.5122 | 176.681 | 19.4644 | 26.376 |
| 50 | 1 | 70 | 26.8114 | 59.4153 | 72.4445 | 203.796 | 19.4116 | 26.3525 |
| 50 | 1 | 80 | 24.7259 | 59.3868 | 83.8295 | 233.17 | 19.3922 | 26.305 |
| 50 | 1 | 90 | 24.3805 | 59.0522 | 92.7387 | 260.98 | 19.4507 | 26.2841 |
| 50 | 1 | 100 | 24.5236 | 59.159 | 103.806 | 290.944 | 19.4324 | 26.2787 |