I start testing with b = 6, v = 4. Because 98304 = 2^15 + 2^16. There are three possibilities:

1. Make C = 16, c = 14, then we need to make k = 3 because of overheads.
2. Make C = 16, c = 13, then we can make k = 4.
3. Make C = 15, c = 15, then we can make k = 4.

I test the possible combinations of (c, C, k) to be (14, 16, 3), (13, 16, 4), and (15, 15, 4).

**astar.trace**

I vary the possible combinations of (s, S) and find the best configuration for each (c, C, k).

(c, C, k) = (14, 16, 3)

| S | S | AAT (s) |
|---|---|---|
| 0 | 0 | 4.116251 |
| 0 | 1 | 4.018445 |
| 0 | 2 | 3.997215 |
| 0 | 3 | 4.027672 |
| 0 | 4 | 4.084612 |
| 1 | 0 | 4.107232 |
| 1 | 1 | 4.107760 |
| 1 | 2 | 4.116425 |
| 1 | 3 | 4.140404 |
| 1 | 4 | 4.194694 |
| 2 | 0 | 4.277815 |
| 2 | 1 | 4.286252 |
| 2 | 2 | 4.302348 |
| 2 | 3 | 4.326738 |
| 2 | 4 | 4.377133 |

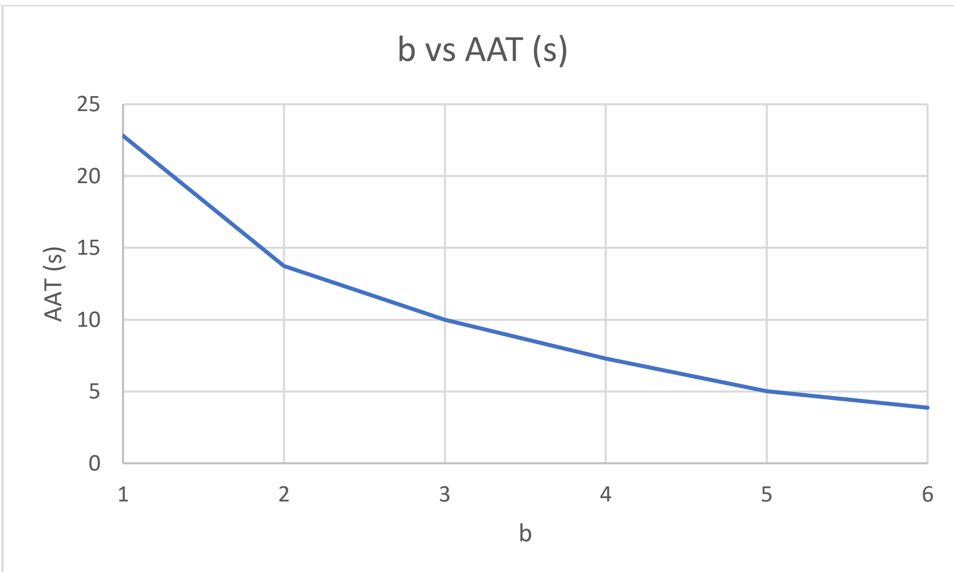
(c, C, k) = (13, 16, 4)

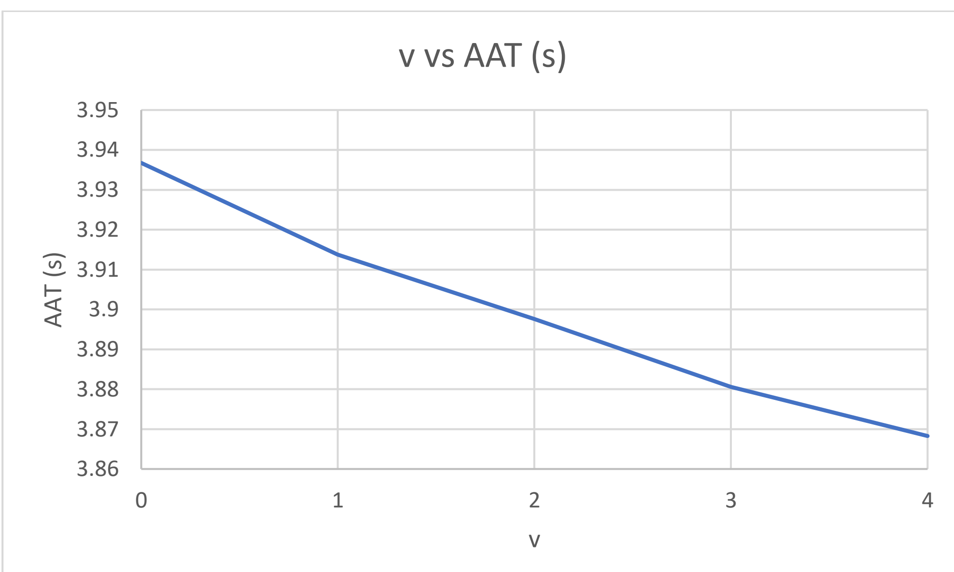
| S | S | AAT (s) |
|---|---|---|
| 0 | 0 | 4.070146 |
| 0 | 1 | 3.892439 |
| 0 | 2 | 3.877933 |
| 0 | 3 | 3.912881 |
| 0 | 4 | 3.975269 |
| 1 | 0 | 3.999852 |
| 1 | 1 | 3.962857 |
| 1 | 2 | 3.970691 |
| 1 | 3 | 3.997668 |
| 1 | 4 | 4.055276 |
| 2 | 0 | 4.163447 |
| 2 | 1 | 4.133159 |
| 2 | 2 | 4.143711 |
| 2 | 3 | 4.173088 |
| 2 | 4 | 4.231180 |

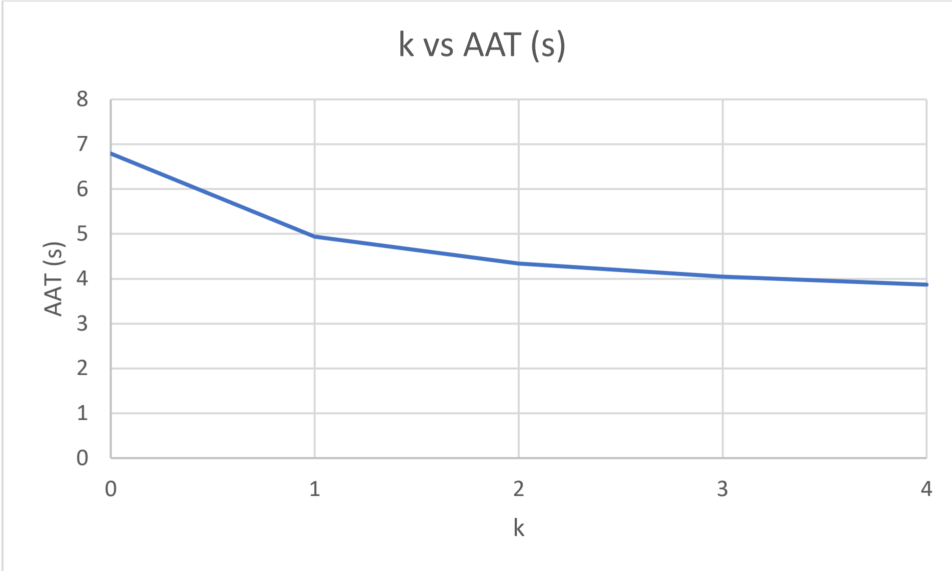
(c, C, k) = (15, 15, 4)

| S | S | AAT (s) |
|---|---|---|
| 0 | 0 | 3.879474 |
| 0 | 1 | 3.868276 |
| 0 | 2 | 3.884040 |
| 0 | 3 | 3.944950 |
| 0 | 4 | 4.030110 |
| 1 | 0 | 3.929275 |
| 1 | 1 | 3.999650 |
| 1 | 2 | 4.038916 |
| 1 | 3 | 4.109770 |
| 1 | 4 | 4.194503 |
| 2 | 0 | 4.104531 |
| 2 | 1 | 4.169115 |
| 2 | 2 | 4.239441 |
| 2 | 3 | 4.308172 |
| 2 | 4 | 4.391101 |

Then we fix c = 15, C = 15, s = 0, and S = 1 to find optimal b, v, and k.






As the graphs show, the larger b, v, and k are, the smaller AAT is.
Therefore, the best configuration is c = 15, C = 15, s = 0, S = 1, b = 6, v = 4, k = 4. AAT = 3.868276 s.
Total budget = 2^15 + 2^15 + (2^9 x (64 – 15 + 0 + 2)) / 8 + (2^9 x (64 – 15 + 1 + 2)) / 8 + 4 x 2048 + 2^6 x 4 + (4 x (64 – 6 + 2)) / 8 = 80606 bytes < 98304 bytes.

**bzip2.trace**

I vary the possible combinations of (s, S) and find the best configuration for each (c, C, k).

(c, C, k) = (14, 16, 3)

| S | S | AAT (s) |
|---|---|---|
| 0 | 0 | 2.086169 |
| 0 | 1 | 2.070702 |
| 0 | 2 | 2.070367 |
| 0 | 3 | 2.071943 |
| 0 | 4 | 2.073666 |
| 1 | 0 | 2.253934 |
| 1 | 1 | 2.251944 |
| 1 | 2 | 2.250101 |
| 1 | 3 | 2.250756 |
| 1 | 4 | 2.251557 |
| 2 | 0 | 2.446353 |
| 2 | 1 | 2.445968 |
| 2 | 2 | 2.446610 |
| 2 | 3 | 2.447106 |
| 2 | 4 | 2.447749 |

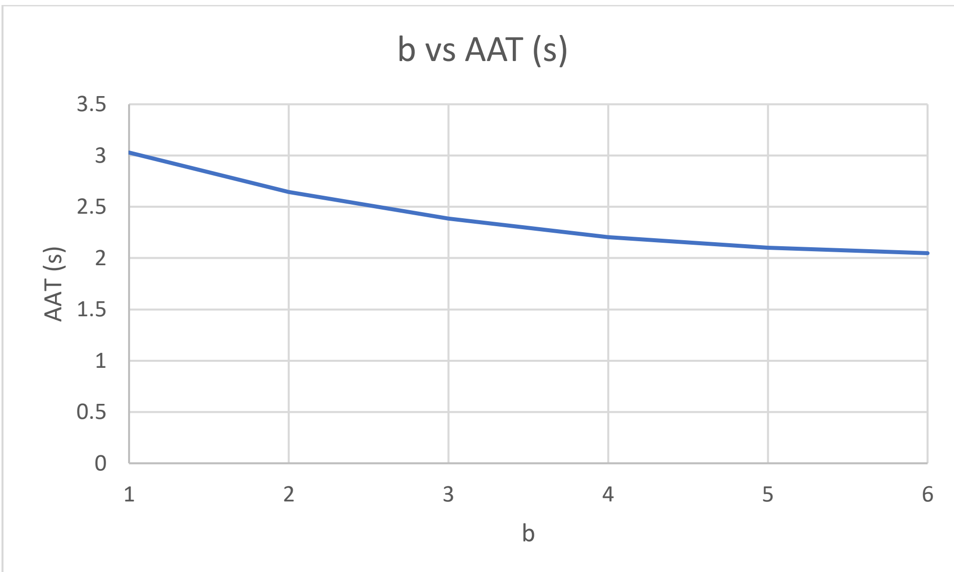
(c, C, k) = (13, 16, 4)

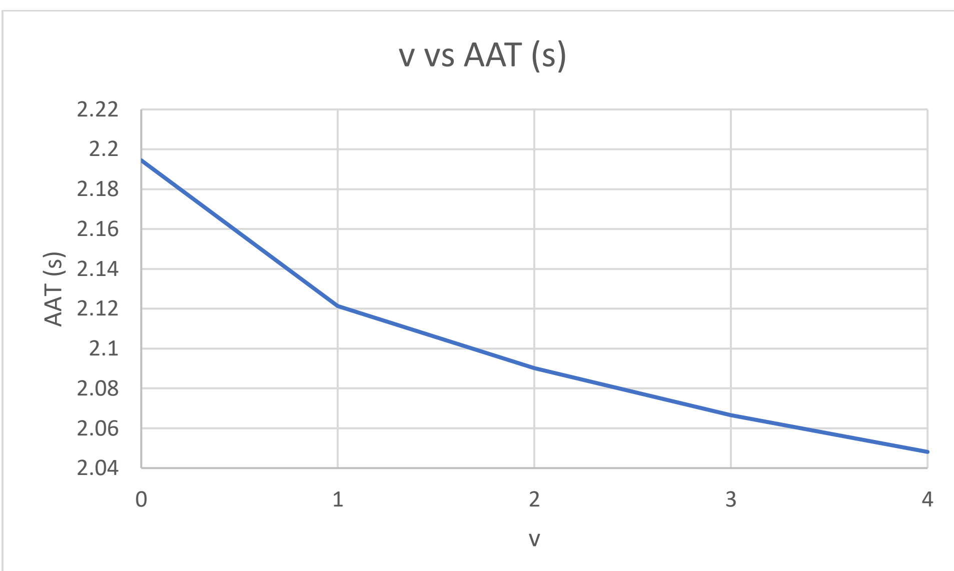
| S | S | AAT (s) |
|---|---|---|
| 0 | 0 | 2.089225 |
| 0 | 1 | 2.070663 |
| 0 | 2 | 2.070614 |
| 0 | 3 | 2.072475 |
| 0 | 4 | 2.074483 |
| 1 | 0 | 2.256755 |
| 1 | 1 | 2.252460 |
| 1 | 2 | 2.251544 |
| 1 | 3 | 2.252538 |
| 1 | 4 | 2.253679 |
| 2 | 0 | 2.458283 |
| 2 | 1 | 2.454821 |
| 2 | 2 | 2.453710 |
| 2 | 3 | 2.454802 |
| 2 | 4 | 2.456041 |

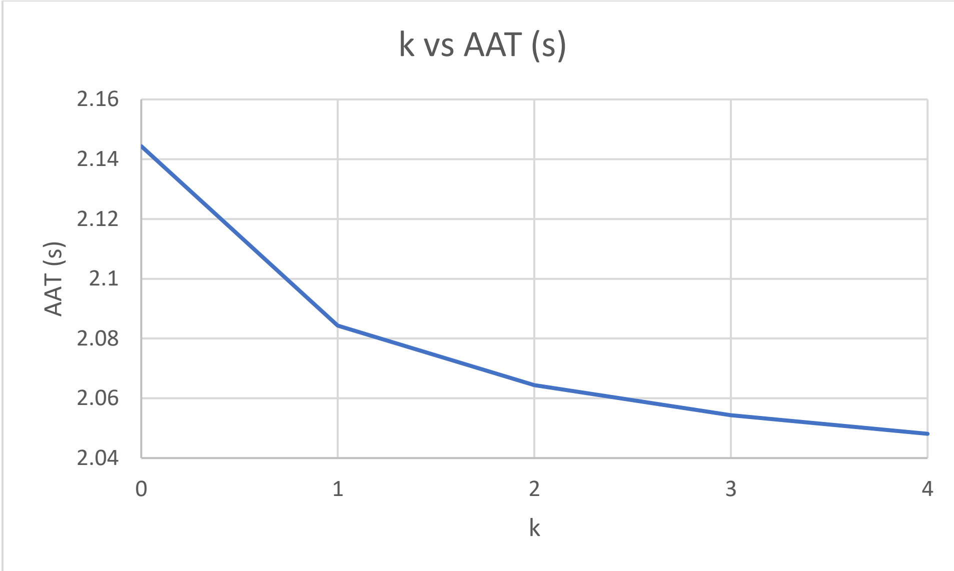
(c, C, k) = (15, 15, 4)

| S | S | AAT (s) |
|---|---|---|
| 0 | 0 | 2.053009 |
| 0 | 1 | 2.049716 |
| 0 | 2 | 2.048481 |
| 0 | 3 | 2.048127 |
| 0 | 4 | 2.048801 |
| 1 | 0 | 2.246015 |
| 1 | 1 | 2.246450 |
| 1 | 2 | 2.246004 |
| 1 | 3 | 2.244088 |
| 1 | 4 | 2.245257 |
| 2 | 0 | 2.439683 |
| 2 | 1 | 2.439721 |
| 2 | 2 | 2.440493 |
| 2 | 3 | 2.441118 |
| 2 | 4 | 2.441596 |

Then we fix c = 15, C = 15, s = 0, and S = 3 to find optimal b, v, and k.






As the graphs show, the larger b, v, and k are, the smaller AAT is.
Therefore, the best configuration is c = 15, C = 15, s = 0, S = 3, b = 6, v = 4, k = 4. AAT = 2.048127 s.
Total budget = 2^15 + 2^15 + (2^9 x (64 – 15 + 0 + 2)) / 8 + (2^9 x (64 – 15 + 3 + 2)) / 8 + 4 x 2048 + 2^6 x 4 + (4 x (64 – 6 + 2)) / 8 = 80734 bytes < 98304 bytes.

**mcf.trace**

I vary the possible combinations of (s, S) and find the best configuration for each (c, C, k).

(c, C, k) = (14, 16, 3)

| S | S | AAT (s) |
|---|---|---|
| 0 | 0 | 2.757849 |
| 0 | 1 | 2.631125 |
| 0 | 2 | 2.607768 |
| 0 | 3 | 2.627114 |
| 0 | 4 | 2.654812 |
| 1 | 0 | 2.642450 |
| 1 | 1 | 2.661870 |
| 1 | 2 | 2.654976 |
| 1 | 3 | 2.668251 |
| 1 | 4 | 2.689877 |
| 2 | 0 | 2.768800 |
| 2 | 1 | 2.804810 |
| 2 | 2 | 2.809620 |
| 2 | 3 | 2.821205 |
| 2 | 4 | 2.840827 |

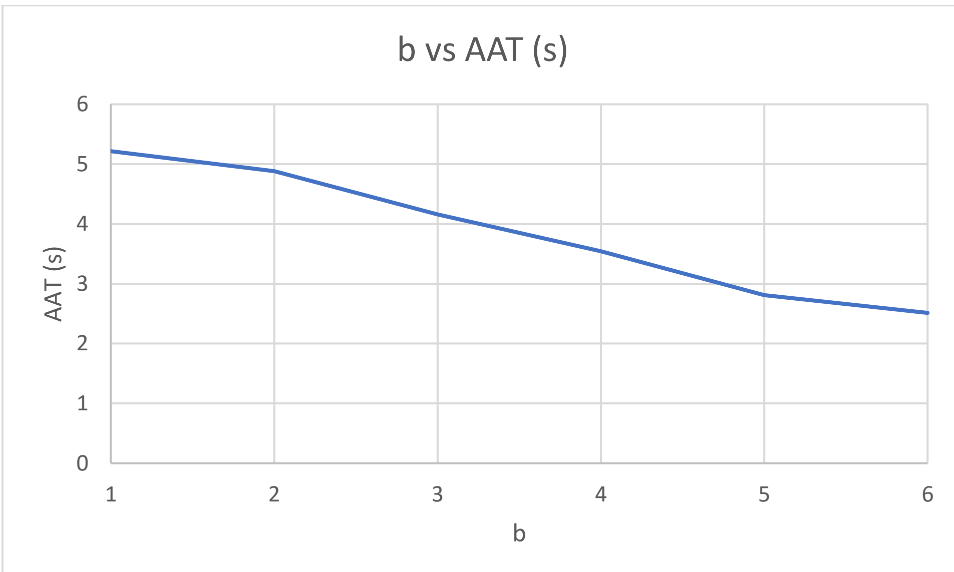
(c, C, k) = (13, 16, 4)

| S | S | AAT (s) |
|---|---|---|
| 0 | 0 | 2.980926 |
| 0 | 1 | 2.864126 |
| 0 | 2 | 2.852270 |
| 0 | 3 | 2.880753 |
| 0 | 4 | 2.924206 |
| 1 | 0 | 2.663155 |
| 1 | 1 | 2.704166 |
| 1 | 2 | 2.697747 |
| 1 | 3 | 2.711498 |
| 1 | 4 | 2.740061 |
| 2 | 0 | 2.758243 |
| 2 | 1 | 2.819937 |
| 2 | 2 | 2.821280 |
| 2 | 3 | 2.832236 |
| 2 | 4 | 2.855797 |

(c, C, k) = (15, 15, 4)

| S | S | AAT (s) |
|---|---|---|
| 0 | 0 | 2.552846 |
| 0 | 1 | 2.549250 |
| 0 | 2 | 2.513508 |
| 0 | 3 | 2.538747 |
| 0 | 4 | 2.583998 |
| 1 | 0 | 2.665424 |
| 1 | 1 | 2.714443 |
| 1 | 2 | 2.698542 |
| 1 | 3 | 2.717621 |
| 1 | 4 | 2.758131 |
| 2 | 0 | 2.806571 |
| 2 | 1 | 2.819415 |
| 2 | 2 | 2.885518 |
| 2 | 3 | 2.905138 |
| 2 | 4 | 2.939254 |

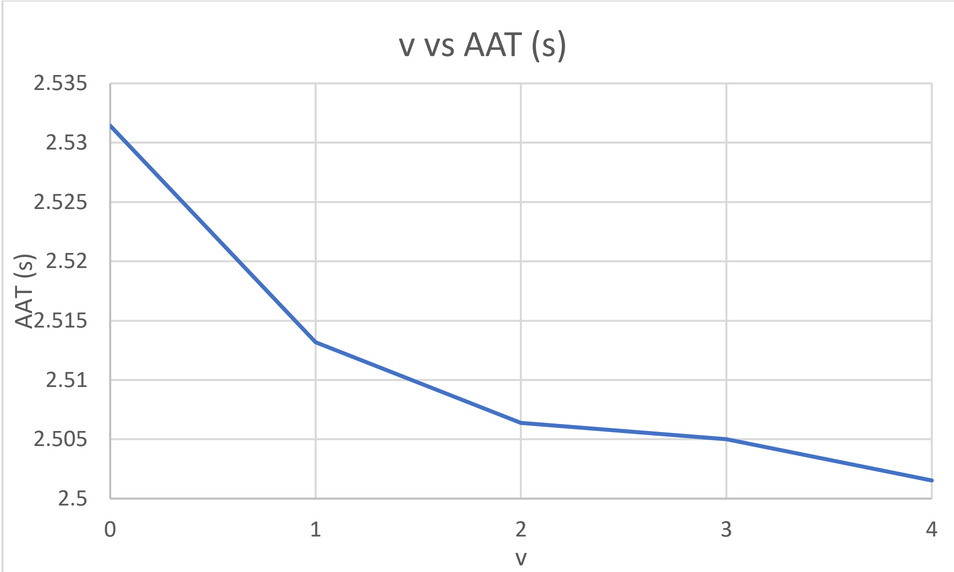
Then we fix c = 15, C = 15, s = 0, and S = 2 to find optimal b, v, and k.




In fact, k = 4 will overexploit spatial locality and result in pollution. k = 3 is the best option here.

| k | AAT (s) |
|---|---|
| 0 | 3.013803 |
| 1 | 2.707323 |
| 2 | 2.522490 |
| 3 | 2.501532 |
| 4 | 2.513508 |

We choose k = 3 and determine v in the final step.




As the graphs show, as the larger b and v are, the smaller AAT is. But k = 4 will result in slight pollution, so we choose k = 3.

Therefore, the best configuration is c = 15, C = 15, s = 0, S = 2, b = 6, v = 4, k = 3. AAT = 2.501532 s.

Total budget = 2^15 + 2^15 + (2^9 x (64 – 15 + 0 + 2)) / 8 + (2^9 x (64 – 15 + 2 + 2)) / 8 + 3 x 2048 + 2^6 x 4 + (4 x (64 – 6 + 2)) / 8 = 78622 bytes < 98304 bytes.

**perlbench.trace**

I vary the possible combinations of (s, S) and find the best configuration for each (c, C, k).

(c, C, k) = (14, 16, 3)

| S | S | AAT (s) |
|---|---|---|
| 0 | 0 | 3.999886 |
| 0 | 1 | 3.782838 |
| 0 | 2 | 3.646194 |
| 0 | 3 | 3.630943 |
| 0 | 4 | 3.698776 |
| 1 | 0 | 4.000044 |
| 1 | 1 | 3.858019 |
| 1 | 2 | 3.790249 |
| 1 | 3 | 3.769145 |
| 1 | 4 | 3.847835 |
| 2 | 0 | 3.988819 |
| 2 | 1 | 3.941238 |
| 2 | 2 | 3.927869 |
| 2 | 3 | 3.917494 |
| 2 | 4 | 3.972389 |

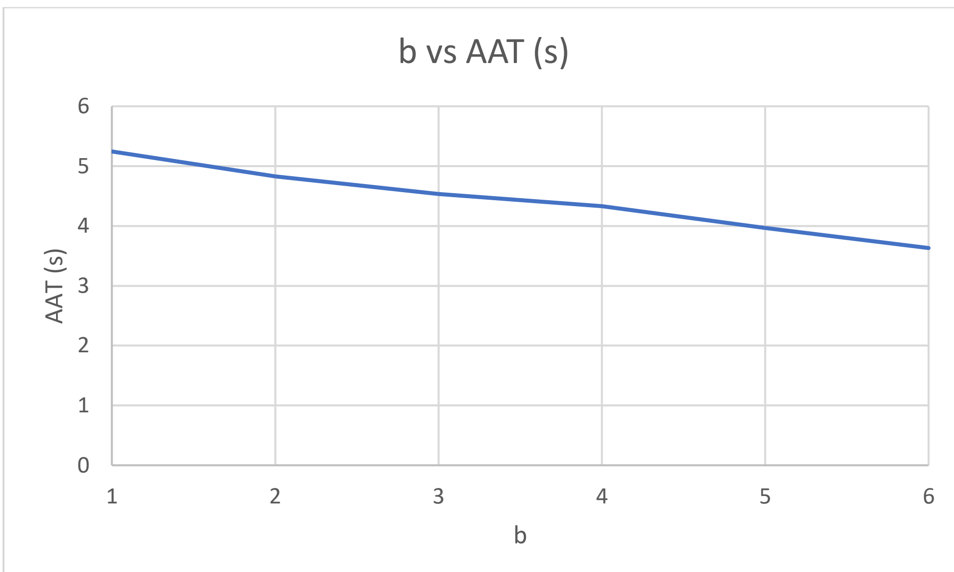
(c, C, k) = (13, 16, 4)

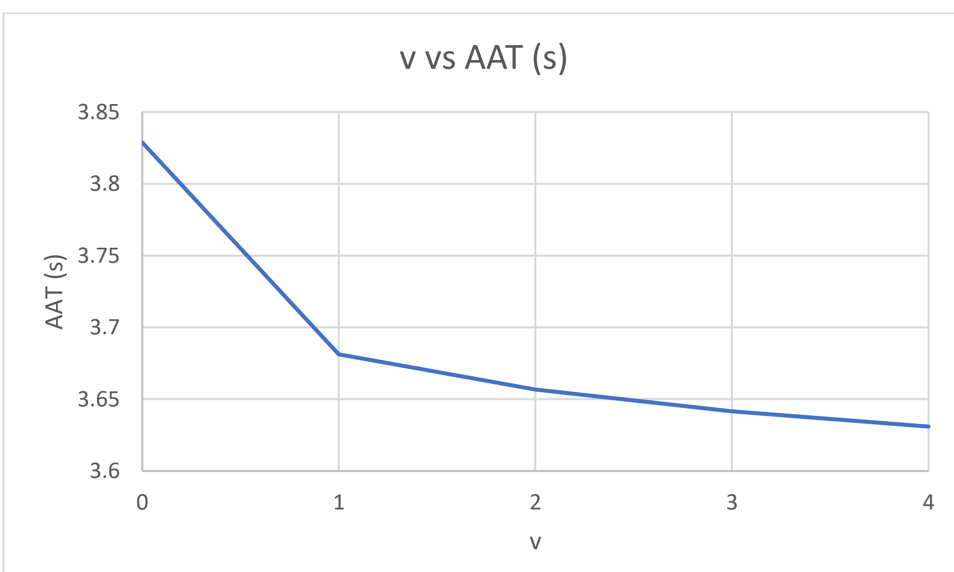
| S | S | AAT (s) |
|---|---|---|
| 0 | 0 | 4.422177 |
| 0 | 1 | 3.990378 |
| 0 | 2 | 3.832267 |
| 0 | 3 | 3.836538 |
| 0 | 4 | 3.925470 |
| 1 | 0 | 4.332583 |
| 1 | 1 | 4.031023 |
| 1 | 2 | 3.901778 |
| 1 | 3 | 3.896765 |
| 1 | 4 | 3.968213 |
| 2 | 0 | 4.448207 |
| 2 | 1 | 4.163183 |
| 2 | 2 | 4.051578 |
| 2 | 3 | 4.034093 |
| 2 | 4 | 4.104263 |

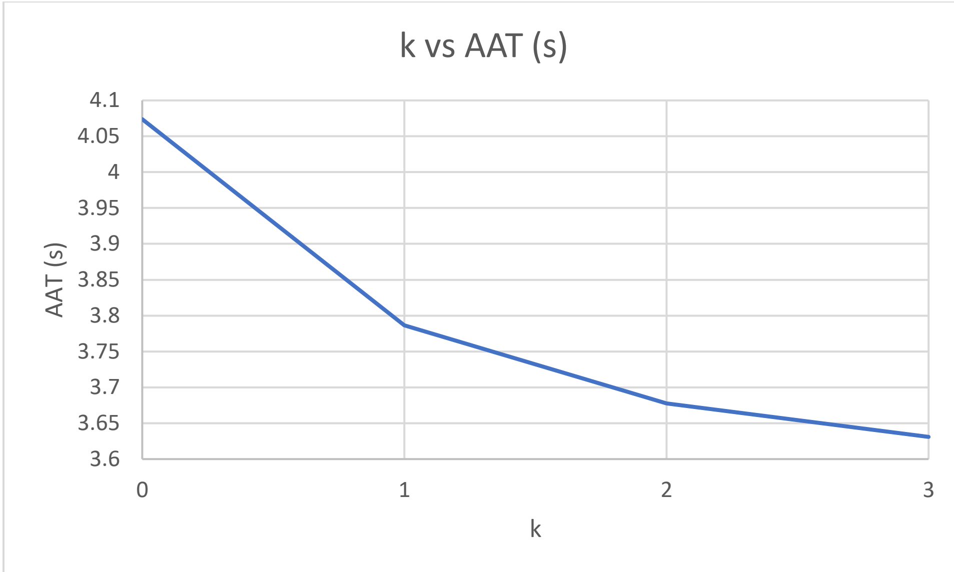
(c, C, k) = (15, 15, 4)

| S | S | AAT (s) |
|---|---|---|
| 0 | 0 | 4.031732 |
| 0 | 1 | 3.943684 |
| 0 | 2 | 3.860997 |
| 0 | 3 | 3.836327 |
| 0 | 4 | 3.884335 |
| 1 | 0 | 3.872092 |
| 1 | 1 | 3.977232 |
| 1 | 2 | 3.992509 |
| 1 | 3 | 3.998485 |
| 1 | 4 | 4.060271 |
| 2 | 0 | 3.973700 |
| 2 | 1 | 4.060457 |
| 2 | 2 | 4.195929 |
| 2 | 3 | 4.183837 |
| 2 | 4 | 4.251991 |

Then we fix c = 14, C = 16, s = 0, and S = 3 to find optimal b, v, and k.






As the graphs show, the larger b, v, and k are, the smaller AAT is.
The best configuration is c = 14, C = 16, s = 0, S = 3, b = 6, v = 4, k = 3. AAT = 3.630943 s.
Total budget = 2^14 + 2^16 + (2^8 x (64 – 14 + 0 + 2)) / 8 + (2^10 x (64 – 16 + 3 + 2)) / 8 + 3 x 2048 + 2^6 x 4 + (4 x (64 – 6 + 2)) / 8 = 96798 bytes < 98304 bytes.