

Approximation Approach

It is greedy on the choice of picking the bin that will leave the least amount of room left in the bin.

If it doesn't fit in any bin then it adds a new one and add it to that.

The randomness comes in that every time it shuffles the item randomly beforehand.

This allows for it to be an anytime solution and stops the greedy solution from getting stuck in bad item order.

Example of greedy not getting the solution immediately:

Bin Capacity: 10

Items: [2,5,4,8]

Optimal is: [[8, 2], [5, 4]]

But we get in this item ordering: [[2, 5], [4], [8]]

Runtime Analysis

Shuffling the items is $O(n)$

For each item (n items):

- Checks all existing bins to find the best-fit

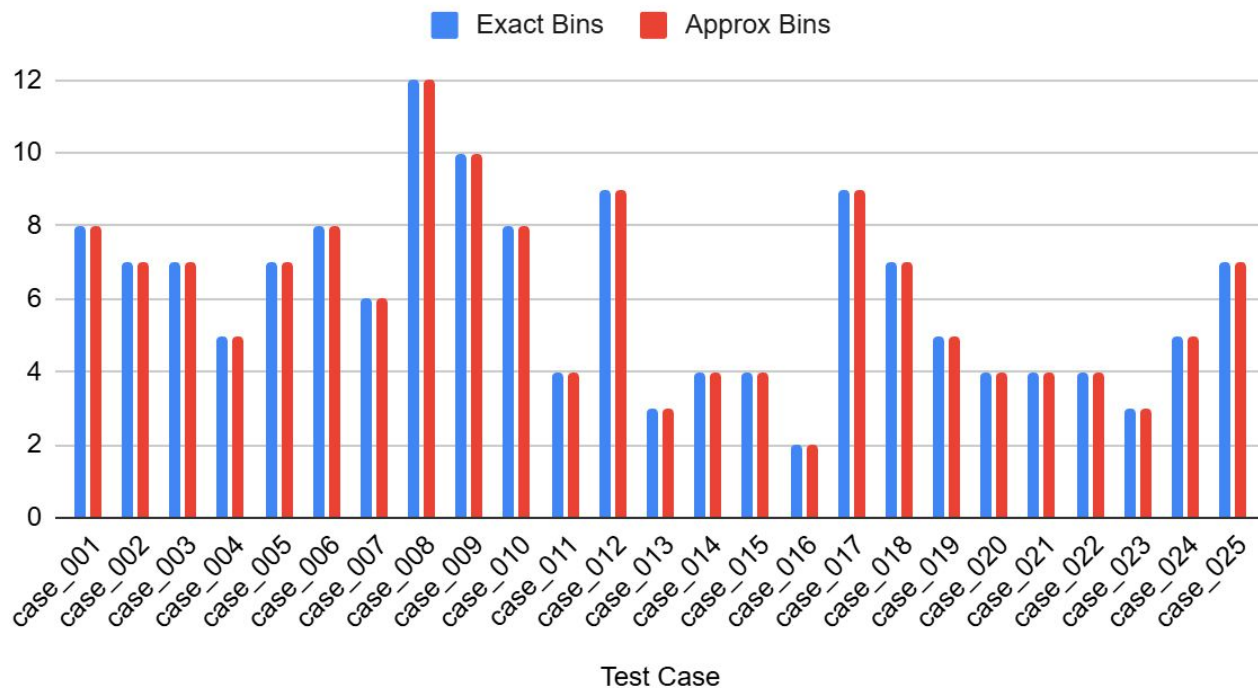
- b = number of bins currently being used

- Worst case $b = n$ (every item has its own bin)

So it is $O(b \cdot n)$

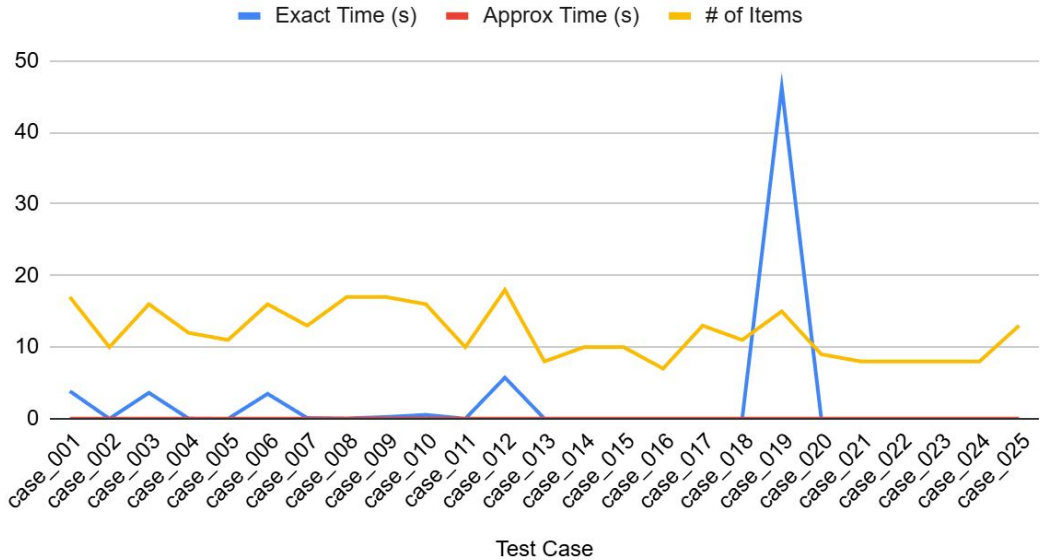
Worst case $O(n^2)$

Optimal vs Approximate Solutions

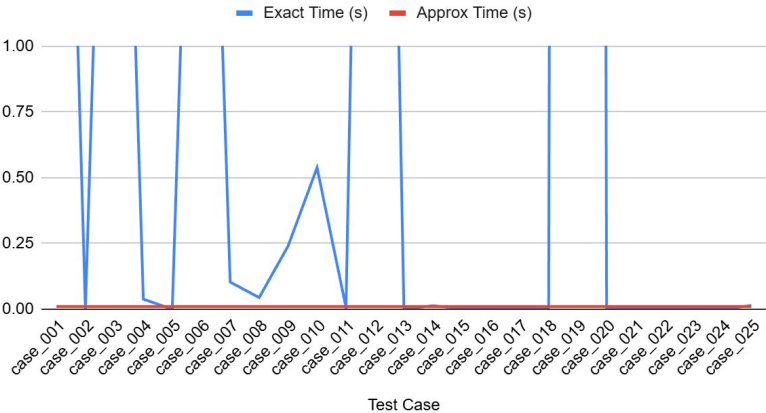


Approximate gets the optimal solution every time with a time limit of 0.01

Optimal vs Approximate Solutions



Optimal vs Approximate Solutions



Really Long Exact Solution vs Approximate Solution

Problem

Items: [99635640, 63621502, 902018,
47517191, 39579336, 10114865,
63121159, 69540985, 45774573,
52693192, 85931861, 59354337,
89422321, 91101777, 42188563,
98364729, 17346065, 6796916,
56849433, 2397607, 74115206,
20816042, 86930618, 35282076,
9937176, 50563, 6845652] (len = 27)

Bin Capacity: 100,000,000

Exact Solution

Optimal bins: 14

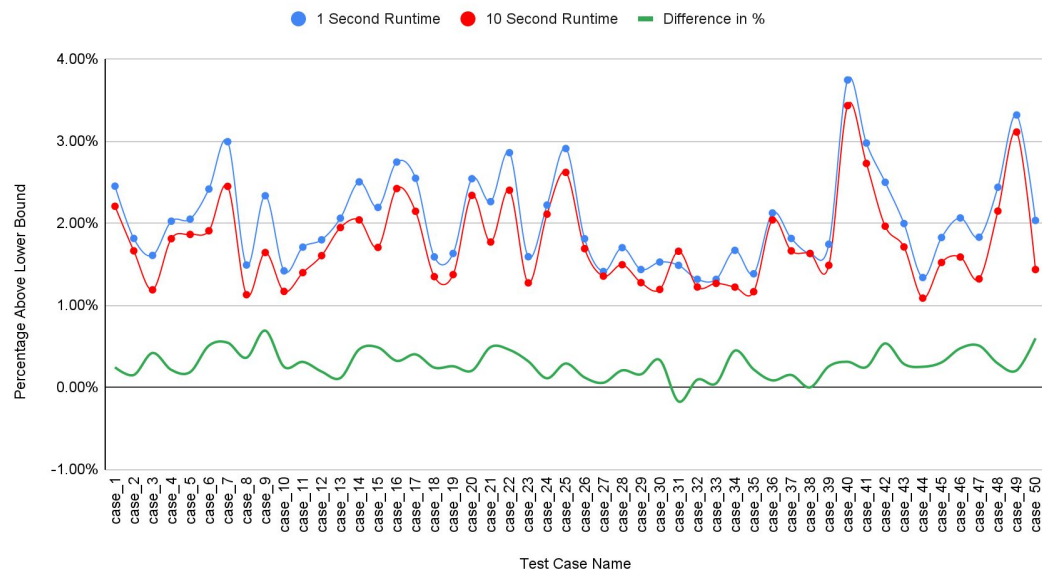
Time: 3804.051s

Approx Solution

Finds 15 Bins in 0.142s

Finds 14 Bins in 0.285s

Percentage above Lower Bound per Test Case (Lower Bound = Sum of Items / Bin Capacity)



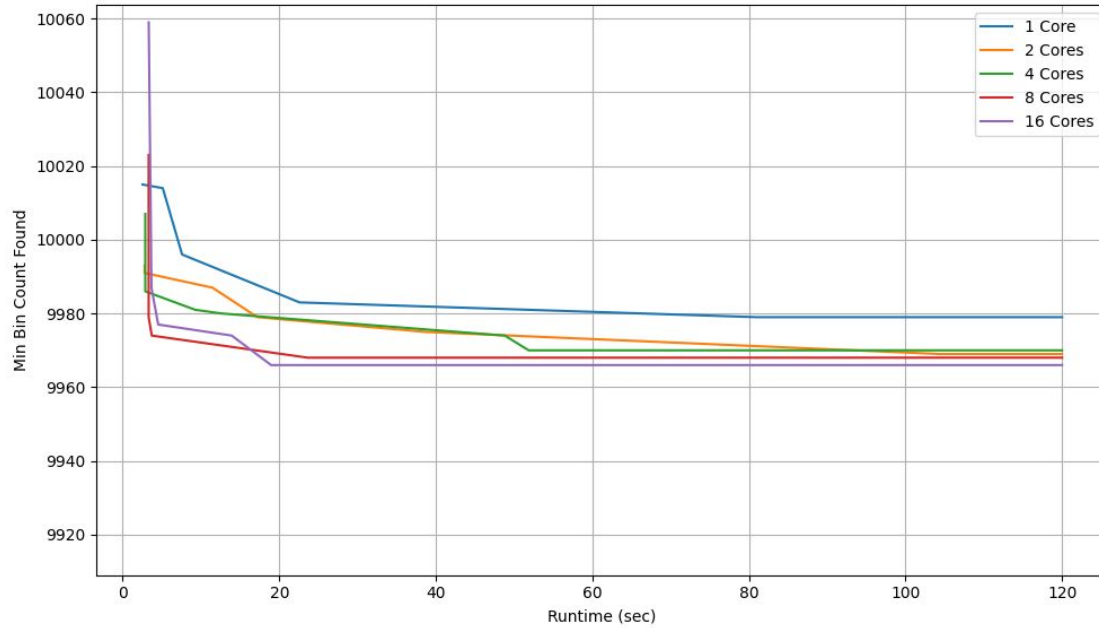
Worst: case_31: -0.17%

Best: case_9: 0.69%

Average: 0.29%

I ran case_31 again until it got something better than the 1 sec Runtime approximation of 1774 Bins. It got 1772 Bins at 145.508s

Bin Solution vs Runtime for Each Process Count



The more cores the faster it goes down and the earlier it finds its final solution. This auto-terminated after 120 seconds.

The lower bound of the Y axis is also the lower bound of the items set for reference (9909).