Homework 3

Adam Karl

September 8, 2020

1 Problem 7 (2 points)

1.1 a

Suppose there are two files with (length, access probability) ordered pairs of (1,0) and (2,1).

The shortest first algorithm places the (1,0) file first on the tape, and the (2,1) file right after, giving an expected access time of $0(0) + 1(1) = 1$.

An optimal solution would place (2,1) (the file accessed 100% of the time) first, and have an expected access time of $1(0) + 0(2) = 0$. Note that this is less than the algorithm's expected access time of 1.

Since we have found a single input where shortest file first does not minimize expected access time, we may conclude (proof by counterexample) that shortest file first is not an optimal algorithm.

1.2 b

Suppose there are two files with (length, access probability) ordered pairs of (100,.6) and (1,.4).

The most likely accessed first algorithm places the (100,.6) file first on the tape, and the (1, .4) file right after, giving an expected access time of $.6(0) + .4(100) = 40$.

An optimal solution would place (1,.4) first, followed by (100,.6) and have an expected access time of $.4(0) + .6(1) = 0.6$. Note that this is less than the algorithm's expected access time of 40.

Since we have found a single input where the most likely accessed first algorithm does not minimize expected access time, we may conclude (proof by counterexample) that the most likely accessed first algorithm is not an optimal algorithm.

1.3 c

For this problem I will refer to the algorithm detailed in (c) as Best Ratio First, or BRF.

Approach: proof by contradiction using exchange argument

- Assume Best Ratio First is not an correct
- therefore, an input I exists such that BRF(I) is not an optimal solution
- let OPT(I) be an optimal solution for input I that has the same file on the tape as BRF(I) for the longest time (starting at the beginning of the tape)
- let u be the first point that BRF(I) and OPT(I) disagree (have different files on the tape).

there are only 2 logical options case 1: OPT(I) does not have a file at point u, but BRF(I) does. Note that the reverse cannot happen since by definition of BRF will always have files on the tape in immediate succession until there are no files left.

- case 1a: some files after point u of OPT(I) have an access probability greater than zero.
  - let v be the length of the gap in OPT(I) from u to the beginning of the first file after u on the tape
  - create OPT'(I) by taking OPT(I) and simply shifting all files toward the front of the tape by a length equal to the gap from u to the first
  - since some of the shifted files had positive access probabilities, OPT'(I) is a strictly better solution than OPT(I) since it decreases the length to the start of these file(s)

- thus, OPT(I) is not an optimal solution since OPT'(I) is a better solution
- since OPT(I) was defined as an optimal solution, and we found that OPT(I) is not an optimal solution for this case, we have a contradiction
- thus, our (only) assumption that BRT is not an optimal solution is incorrect
- thus, BRT is an optimal solution

- case 1a: all files after point u of OPT(I) have an access probability greater than zero.
  - let x be the file BRT(I) has at u
  - create OPT'(I) by taking OPT(I) and simply shifting all files toward the back of the tape by the length of x
  - now, find x on OPT'(I) and move it to start at u. Note that we are certain it does not overlap any other files since we first made a gap such that x could fit at u perfectly
  - since no files overlap, OPT'(I) is feasible
  - since all files after u have an access probability of 0, the average access time of OPT'(I) is exactly equal to OPT(I)
  - thus, OPT'(I) is at least as good of a solution as OPT(I)
  - thus, OPT'(I) is at least as good of a solution as OPT(I), OPT'(I) is feasible, and OPT'(I) agrees with BRF(I) for 1 more step than OPT(I)
  - thus, OPT'(I) is not the optimal solutiom that agrees with BRF(I) for the most number of steps
  - since OPT(I) was defined as the optimal solution that agreed with BRF(I) for the most number of steps, and we found that OPT(I) is NOT the optimal solution that agreed with BRF(I) for the most number of steps, we have a contradiction
  - thus, our (only) assumption that BRT is not an optimal solution is incorrect
  - thus, BRF is an optimal solution

case 2: OPT(I) and BRF(I) both have files at point u, but have different files.

- let x be the file BRF(I) has at u,
- let y_1 through y_n be all the files OPT(I) has after u but before x (since they have the same files up to u, OPT(I) must have x at some point after u, and has at least 1 file between the two)
- start with OPT'(I) = OPT(I) and follow these steps on OPT"(I)
  - swap x with the file that immediately precedes x. We will call this file y_i.
  - let the length of x be l_x and the length of y_i be l_y
  - since x and y_i still take up the same total space, and only switched positions, no files overlap (since OPT(I) started with no 2 files overlapping in order to be feasible)
  - x decreased it's position by l_y
  - y_i increased it's position by l_x
  - thus, the total IMPROVEMENT in avg access time is equal to $x * l_y - y * l_x$
  - thanks to the definition of BRF and the fact that BRF chose x at point u, we know l_x/x is smaller than (or equal to) the corresponding length / access probability for any other file after u.
  - thus, $l_x/x \leq l_y/y$
  - multiply both sides by xy to get $y * l_x \leq x * l_y$
  - since the avg access time improves by $x * l_y - y * l_x$ and $l_x * y \leq l_y * x$, and all values (l_x, l_y, x, y) are nonnegative, OPT'(I) improves by a non-negative amount
  - thus, OPT'(I) is at least as good of a solution as OPT'(I)
- repeat those steps, swapping x one file closer to u at a time until file x is at point u

- since each swap is feasible and makes OPT'(I) no worse of a solution than OPT(I), OPT'(I) is now the optimal solution that agrees with BRF(I).
- thus, OPT'(I) is not the optimal solution that agrees with BRF(I) for the most number of steps
- since OPT(I) was defined as the optimal solution that agreed with BRF(I) for the most number of steps, and we found that OPT(I) is NOT the optimal solution that agreed with BRF(I) for the most number of steps, we have a contradiction
- thus, our (only) assumption that BRT is not an optimal solution is incorrect
- thus, BRF is an optimal solution