Review & Plan
Today’s objectives

- Competitive ratio of Next Fit (Review)
- Competitive ratio of Harmonic
  - Weighting technique for upper bound
  - Lower bound!
- A sketch of competitive ratio of Best Fit and First Fit
Competitive Analysis of Next Fit

- Competitive ratio of NextFit is at most 2.
  - In the final packing, total size of items in each two consecutive bins is larger than 1.
  - The cost of NextFit for serving $\sigma$ is smaller than $2S(\sigma)$ where $S(\sigma)$ is the total size of items in $\sigma$.
    - Assume $\text{cost}(\text{NextFit}) = k$
    - Each two consecutive bins have total size $> 1 \rightarrow$ total-size $S(\sigma)$ of items in $\sigma$ is more than $k/2$
  - $\text{OPT}(\sigma) \geq S(\sigma)$: Even when $\text{OPT}$ packs items tightly (with no wasted space), $S(\sigma)$ bins are required.

![Diagram](image_url)

- Competitive ratio of NextFit is at most 2.
Competitive Analysis of Next Fit

- Competitive ratio of NextFit is at least 2.
  - Consider sequence $\sigma = <0.5, \epsilon, 0.5, \epsilon, \ldots >$.
  - The cost of NextFit for serving $\sigma$ is roughly $n/2$ ($n$ is the length of $\sigma$).
  - The cost of OPT is roughly $n/4$.

\[
\begin{array}{ccccccc}
\epsilon & \epsilon & \epsilon & \epsilon & \epsilon & \epsilon \\
5 & 5 & 5 & 5 & 5 & 5 \\
\end{array}
\]

\[
\begin{array}{cccc}
5 & 5 & 5 \\
5 & 5 & 5 \\
\end{array}
\]

Theorem

Competitive ratio of NextFit is exactly 2.
Weighting Argument for Harmonic Algorithm
Harmonic Algorithm

- Harmonic Algorithm classes: \((\frac{1}{2}, 1], (\frac{1}{3}, \frac{1}{2}], \ldots, (\frac{1}{K}, \frac{1}{K-1}], (0, \frac{1}{K}]\).
- Place members of each class separately from others.

\[
\text{Harmonic } K = 4
\]

< 0.9 0.3 0.8 0.5 0.1 0.1 0.3 0.2 0.4 0.2 0.4 0.5 0.5 0.8 0.6 0.4 0.5 ... >

\[
x > \frac{1}{2}
\]

\[
\frac{1}{3} < x \leq \frac{1}{2}
\]

\[
\frac{1}{4} < x \leq \frac{1}{3}
\]

\[
x \leq \frac{1}{4}
\]
Weighting Technique in a Nutshell

- Assume we want to prove an algorithm Alg is competitive
- Define a **weight** $w(x)$ for each item $x$ based on its size
  - General rule: for an item of size $x$, we should have $w(x) \geq x$
- Weight should be defined so that total weight of items in any bin $B$ of the algorithm (denoted by $w(B)$) is at least 1
  - By ‘any bin’ we mean all bins except possibly a constant number.
  - Assume algorithm opens $k$ bins; we have $k \cdot 1 \leq W$ where $W$ is the total weight of items in the sequence
  - So, we have $\text{Cost}(\text{Alg}) \leq W$ (ignoring a constant no. of bins)
- Find the maximum weight of items that fit in any bin
  - Let $J$ denote that number
  - $\text{OPT}$ has to place items with total weight of $W$ into bins each taking weight $J$ out of it
  - So, we have $\text{Cost}(\text{Opt}) \geq W/J$
- The competitive ratio of the algorithm will be at most $J$
Weighting Technique in a Nutshell

- Step I: Define a weight function $w(x)$ for item sizes
- Step II: Prove that any bin of the online algorithm has weight 1.
- Step III: Prove that it is not possible to place a total weight more than $J$ in any empty bin
- The competitive ratio will be $J$
Weighting Technique

- Define a weight for each item based on its size
- The weight of an item in class \( i \) is \( \frac{1}{i} \) when \( i < k \)
- The weight of an item of size \( x \) in class \( k \) is \( \frac{k}{k-1}x \)

Harmonic \( K = 4 \)

\[
\begin{align*}
&< 0.9 0.3 0.8 0.5 0.1 0.1 0.3 0.2 0.4 0.2 0.4 0.5 0.5 0.8 0.6 0.4 0.5 \ldots > \\
\end{align*}
\]

\[
\begin{array}{cccc}
0.9 & 0.8 & 0.8 & 0.6 \\
0.5 & 0.4 & 0.5 & 0.5 \\
0.3 & 0.3 & 0.2 & 0.2 \\
0.2 & 0.2 & 0.1 & 0.1 \\
\end{array}
\]

- \( x > \frac{1}{2} \) weight = 1
- \( \frac{1}{3} < x \leq \frac{1}{2} \) weight = \( \frac{1}{2} \)
- \( \frac{1}{4} < x \leq \frac{1}{3} \) weight = \( \frac{1}{3} \)
- \( x \leq \frac{1}{4} \) weight = \( \frac{4}{3}x \)
Weighting Technique for Harmonic

- Total weight of items in each bin of Harmonic is at least 1
  - Except possibly the current open bin of each class → k bins
  - Bins of type $i < k$ include $i$ items, each of weight $\frac{1}{i}$ → total weight $i \cdot \frac{1}{i} = 1$
  - Any bin $B$ of type $k$ (except the open bin) has level $> \frac{k-1}{k}$
    - let $y$ be the first item in the next bin opened → $y$ did not fit in the previous bin → level of the bin + size of $y > 1$ $\Rightarrow$ level of $B > \frac{k-1}{k}$.
  - $(\text{Level of } B) > \frac{k-1}{k} \cdot \frac{\text{weight of } x = \frac{k-1}{k} \cdot x}{(\text{total weight of items in } B)} > 1$

\begin{align*}
\text{weight} = 1 & \quad \text{weight} = \frac{1}{2} & \quad \text{weight} = \frac{1}{3} & \quad \text{weight} = \frac{4}{3} x \\
\frac{1}{2} < x & \leq \frac{1}{2} & \frac{1}{3} < x \leq \frac{1}{2} & \frac{1}{4} < x \leq \frac{1}{3} & x & \leq \frac{1}{4}
\end{align*}
Weighting Technique for Harmonic

How much is the maximum total weight of items in a bin of Opt?

- Define density of an item of size $x$ as $\frac{w(x)}{x}$
- Fill the bin with smallest items of classes.
- Use a greedy algorithm that places items with a preference for items of higher density (i.e., larger)!

\[
\begin{array}{cccc}
0.9 & 0.8 & 0.8 & 0.6 \\
x > \frac{1}{2} & & & \\
0.5 & 0.4 & 0.5 & 0.4 \\
\frac{1}{3} < x \leq \frac{1}{2} & & & \\
0.3 & 0.2 & 0.2 & 0.3 \\
\frac{1}{4} < x \leq \frac{1}{3} & & & \\
0.3 & 0.2 & 0.1 & 0.1 \\
x \leq \frac{1}{4} & & & \\
\end{array}
\]

weight = 1 \quad weight = \frac{1}{2} \quad weight = \frac{1}{3} \quad weight = \frac{4}{3} x
\begin{align*}
\rho & \leq 2 \\
\rho & \leq \frac{3}{2} \\
\rho & \leq \frac{4}{3} \\
\rho & = \frac{(k + 1)}{k} = \frac{4}{3} \\
\end{align*}
Weighting Technique for Harmonic

How much is the maximum total weight of items in a bin of Opt?

- Next largest item that fits: $1/2 + \epsilon$; weight = 1; size = $1/2 + \epsilon$
- Next item that fits: $1/3 + \epsilon$; weight = $1 + 1/2 + 1/6$; size = $5/6 + 1/7 + 3\epsilon = 41/42 + 3\epsilon$
- Next item that fits: $1/7 + \epsilon$; weight = $1 + 1/2 + 1/6 + 1/42$; size = $41/42 + 1/43 + 4\epsilon$

$$
\begin{align*}
\text{weight} &= 1 \\
\rho &\leq 2 \\
\text{weight} &= 1/2 \\
\rho &\leq 3/2 \\
\text{weight} &= 1/3 \\
\rho &\leq 4/3 \\
\text{weight} &= 4 \times x \\
\rho &= (k + 1)/k = 4/3
\end{align*}
$$
Weighting Technique for Harmonic

How much is the maximum total weight of items in a bin of Opt?

- So, the greedy approach fills a bin with total weight
  \[ 1 + \frac{1}{2} + \frac{1}{6} + \frac{1}{42} + \frac{1}{(42 \cdot 43)} \ldots \approx 1.691 \]

- It turns out that it is not possible to achieve higher weight
  - E.g., if there is no item of class 1, the density and hence total weight will be less than \( \frac{3}{2} \) → there is an item of size \( \frac{1}{2} + \epsilon \)
  - If there is an item of size \( \frac{1}{2} + \epsilon \) and no item of class 2, there can be at most one item \( \frac{1}{4} + \epsilon \) of class 3, and density of the rest is less than \( \frac{5}{4} \). Weight will be \( 1 + \frac{1}{3} + \frac{5}{4} \cdot \frac{1}{4} \approx 1.64 \) → there is an item of size \( \frac{1}{3} + \epsilon \)

<table>
<thead>
<tr>
<th>( x &gt; \frac{1}{2} )</th>
<th>( \frac{1}{3} &lt; x \leq \frac{1}{2} )</th>
<th>( \frac{1}{4} &lt; x \leq \frac{1}{3} )</th>
<th>( x \leq \frac{1}{4} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>weight = 1</td>
<td>weight = ( \frac{1}{2} )</td>
<td>weight = ( \frac{1}{3} )</td>
<td>weight = ( \frac{4}{3} x )</td>
</tr>
<tr>
<td>( \rho \leq 2 )</td>
<td>( \rho \leq \frac{3}{2} )</td>
<td>( \rho \leq \frac{4}{3} )</td>
<td>( \rho = \frac{k+1}{k} = \frac{4}{3} )</td>
</tr>
</tbody>
</table>
Summary of Weighting Technique for Harmonic

- We define a weight of an item of class $i < k$ to be $1/i$ and the weight of an item of class $k$ to be $\frac{k}{k-1} \cdot x$

- We showed that the weight of all bins (except at most $k$ of them) is at least 1 in Harmonic’s packing.

- We showed that the maximum weight of any bin is at most $J = 1 + \frac{1}{2} + \frac{1}{6} + \frac{1}{42} + \ldots$ when $k$ is large enough.
  - We often assume $k$ is a constant around 20.

- The competitive ratio of the algorithm will be at most $J$
Lower Bound: a Nasty Sequence

- Consider the following sequence

\[
\langle 1/43 + \epsilon, \ldots, 1/43 + \epsilon, 1/7 + \epsilon, \ldots, 1/7 + \epsilon, 1/3 + \epsilon, \ldots, 1/3 + \epsilon, 1/2 + \epsilon, \ldots, 1/2 + \epsilon, \rangle
\]

- Harmonic opens \( m(1/42 + 1/6 + 1/2 + 1) \approx 1.691m \) bins

- OPT places one item of each class in each bin \( \rightarrow m \) bins
Lower Bound: a Nasty Sequence

- Consider the following sequence

\[
\langle \frac{1}{43} + \epsilon, \ldots, \frac{1}{43} + \epsilon, \frac{1}{7} + \epsilon, \ldots, \frac{1}{7} + \epsilon, \frac{1}{3} + \epsilon, \ldots, \frac{1}{3} + \epsilon, \frac{1}{2} + \epsilon, \ldots, \frac{1}{2} + \epsilon, \rangle
\]

- What about First Fit and Best Fit?
- Both create the same packing as Harmonic!
Summary

- Competitive ratio Harmonic is $j = 1.691$.
- Competitive ratios of Best Fit and First Fit is at least $J$
- Indeed their ratio is 1.7
- We see a sketch of the proof in the next class