Announcement
- Groups for Problem Set 1 are assigned on CMS now.
- If you try to contact a group member and they're non-responsive let us know sooner rather than later.
- On future problem sets your group will have the option to remain matched if you all want that.

**Online (Fractional) Matching**

Reminder: $G = (L, R, E)$, $R = [T]$
- When vertex $j \in R$ "arrives" at time $j$,
  the algo learns its set of neighbors, $N(j) \subseteq L$ and needs to:
  - match $j$ with one neighbor or leave it permanently unmatched.
  - (Fractional) choose values $x_{ij} \geq 0$
    such that $\sum_{i \in L} x_{ij} \leq 1$, $x_{ij} = 0$ for $i \notin N(j)$
\[ \sum_{j \in \mathcal{N}_i} x_{ij} \leq 1 \quad \forall i. \]

Naive idea for fractional matching:
- Each \( j \) sets \( x_{ij} = \frac{1}{|\mathcal{N}_i|} \) if \( i \in \mathcal{N}_j \).
- If this doesn't exceed the capacity constraint of any neighbor.

1: splits \( \frac{1}{n+1} \) units to each nbr.
2: same thing
\nFor \( n \) same thing

\( n+1 \): assigns \( \frac{1}{n+1} \) to unique nbr
\( n+2 \): ...

Sum up: vertices in \( R \) assign
\[ n + \frac{n}{n!} < n+1 \]
units, while max matching has size \( 2n \).

"Water-filling algorithm": imagine each \( i \) as a bucket of water. When \( j \) arrives, the level of water in bucket \( i \) is
\[ d_{j-1}(i) = \sum_{k=1}^{j-1} x_{j-1,k} \]

In continuous time, a "scan line" sweeps from level \( \emptyset \) to level 1 and contributes water to buckets where level is below scan line until 1 unit total has been allocated.

\[ x_{j} x_{y} = 0 \quad x_{j} x_{y} \]

\[ x_{j} + x_{y} + x_{z} + x_{y} = 1 \]

Let \( n_{j}(t) = \# \{ i \in N_{j} : d_{j-1}(i) < t \} \).

Let \( \hat{x}(j) \) (the "water level" of \( j \)) be \( \hat{x}(j) \) the unique solution of

\[ \int_{0}^{\hat{x}(j)} n_{j}(t) \, dt = 1. \]

\[ l_{j} = \min \{ a, \hat{x}(j) \}. \]
\[ x_{ij} = \max \left\{ 0, \lambda(j) - \frac{d_{i-1}(i)}{j} \right\} \]

maximize \[ \sum_{(i,j) \in E} x_{ij} \]

subject to \[ \sum_{j} x_{ij} \leq 1 \quad \forall i \] (scale by \( y_i \))

\[ \sum_{i} x_{ij} \leq 1 \quad \forall j \] (scale by \( y_j \))

\[ x_{ij} \geq 0 \quad \forall \ (i,j) \in E \]

\[ \sum_{j} y_i x_{ij} \leq y_i \quad \text{if } y_i \geq 0. \]

\[ \sum_{i} y_j x_{ij} \leq y_j \quad \text{if } y_j \geq 0. \]

\[ \sum_{(i,j) \in E} (y_i + y_j) x_{ij} \leq \sum_{i \in L} y_i + \sum_{j \in R} y_j \]

\[ \sum_{(i,j) \in E} x_{ij} \leq \sum_{i \in L} y_i + \sum_{j \in R} y_j \quad \forall (i,j) \in E. \]

if \( y_i + y_j \geq 1 \)
Plan of attack: enhance WATERFILLING to calculate $y_i$'s and $y_j$'s as it runs, so that in the end:

$$y_i, y_j \geq 0 \quad \forall (i,j)$$

$$y_i + y_j \geq 1 \quad \forall (i,j) \in E$$

$$\sum_{(i,j) \in E} x_{ij} \geq \frac{e-1}{e} \left[ \sum_{i \in E} y_i + \sum_{j \in E} y_j \right]$$

**Proposal:** Let

$$g(z) = \frac{e^z - 1}{e - 1}.$$ 

(1) $g(0) = 0$, $g(1) = 1$

(2) $g$ is increasing on $[0,1]$

(3) $1 - g(z) + g'(z) = \frac{e}{e - 1}$

For all $z \in [0,1]$, $g(z) = \frac{e^z}{e - 1}$

$$1 - g(z) = \frac{e - e^z}{e - 1} \quad g'(z) = \frac{e^z}{e - 1}$$

Let $y_i = g(d(i)) = g\left(\sum_{j} x_{ij}\right)$

$y_j = 1 - g(d(j))$.
\[ y_i - y_j = 0 \]

\[ \forall i, j \in \mathbb{N} \]

As seen in Figure 1, the function \( f \) maps elements to points such that

\[ f(a) = (x, y) \]

Therefore, for any point \( (a, b) \), we have

\[ f(a) + f(b) = f(a+b) \]

\[ \text{If } i, j \in E \text{ we must have } y_i + y_j = 1 \]

\[ d(i) = g(x) + g(y) \]

\[ (d(i) + 0) - 1 = y_i + y_j \]