CLRS3 lemma 24.15 (used in Bellman-Ford). Consider a weighted directed graph. Consider any shortest path from s to t,

$$s = v_0 \to v_1 \to \cdots \to v_k = t.$$

Suppose we initialize the data structure by

v.dist = ∞ for all vertices other than s

s.dist = 0

and then we perform a sequence of relaxation steps that includes, in order, relaxing $v_0 \to v_1$, then $v_1 \to v_2$, then … then $v_{k-1} \to v_k$. After these relaxations, and at all times thereafter, v_k.dist = distance(s to v_k).

We'll prove by induction that, after the ith edge has been relaxed,

v_i.dist $=$ distance(s to v_i)

BASE CASE $i = 0$: Note that $s = v_0$. We initialized s.dist $= 0$, and distance(s to s) $= 0$, so the induction hypothesis is true.

INDUCTION STEP: …

If there's a graph with -ve weight cycle, it's possible that distance (s to s) = -∞!

So, is this proof right, wrong, or not even wrong?

Max-Flow Min-Cut
and
Lagrangian optimization

SECRET RM-1573 10-24-55 -33-

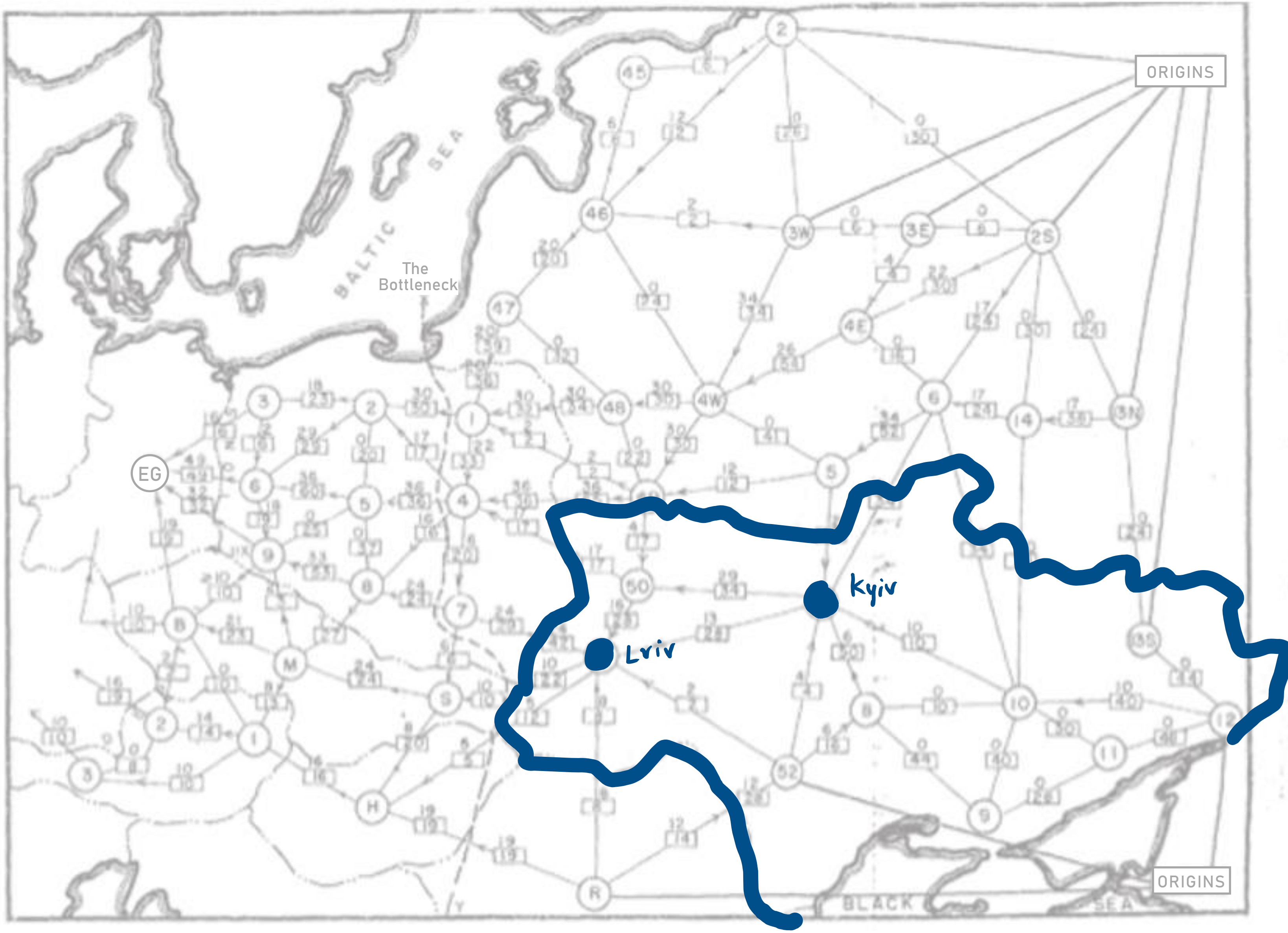


Fig. 7 — Traffic pattern: entire network available

Legend:

International boundary

Railway operating division

Capacity: 12 each way per day. Required flow of 9 per day toward destinations (in direction of arrow) with equivalent number of returning trains in opposite direction

All capacities in trains / 1000's of tons each way per day

Origins: Divisions 2, 3W, 3E, 2S, 13N, 13S, 12, 52 (USSR), and Roumania

Destinations: Divisions 3, 6, 9 (Poland); B (Czechoslovakia); and 2, 3 (Austria)

Alternative destinations: Germany or East Germany

Note IIX of Division 9, Poland

SECRET RM-1573 10-24-55 -33-

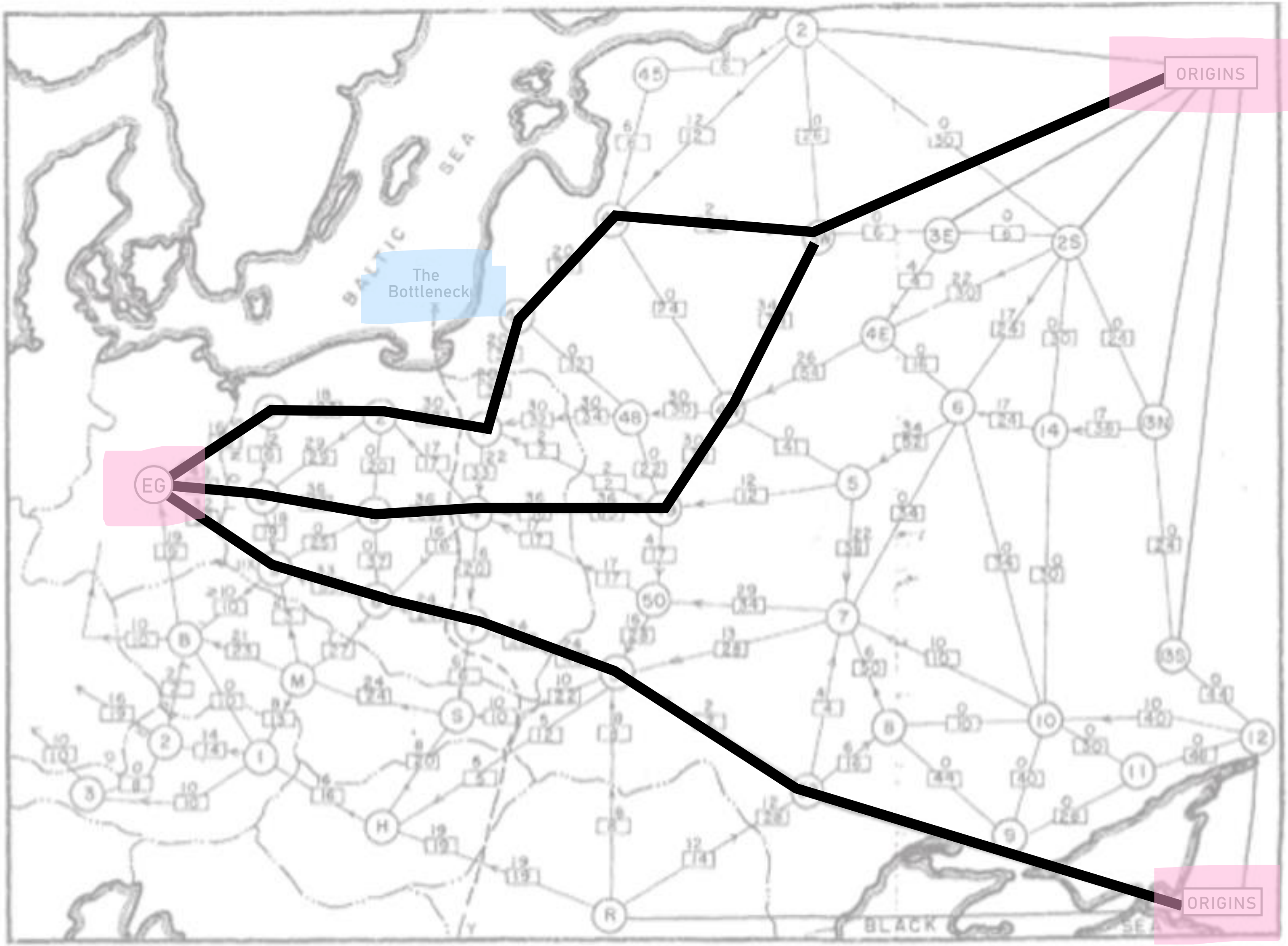


Fig. 7 — Traffic pattern: entire network available

Legend:

International boundary

Railway operating division

Capacity: 12 each way per day. Required flow of 9 per day toward destinations (in direction of arrow) with equivalent number of returning trains in opposite direction

All capacities in trains / 1000's of tons each way per day

Origins: Divisions 2, 3W, 3C, 2S, 13N, 13S, 12, 52 (USSR), and Roumania

Destinations: Divisions 3, 6, 9 (Poland); B (Czechoslovakia); and 2, 3 (Austria)

Alternative destinations: Germany or East Germany

Note IIX of Division 9, Poland

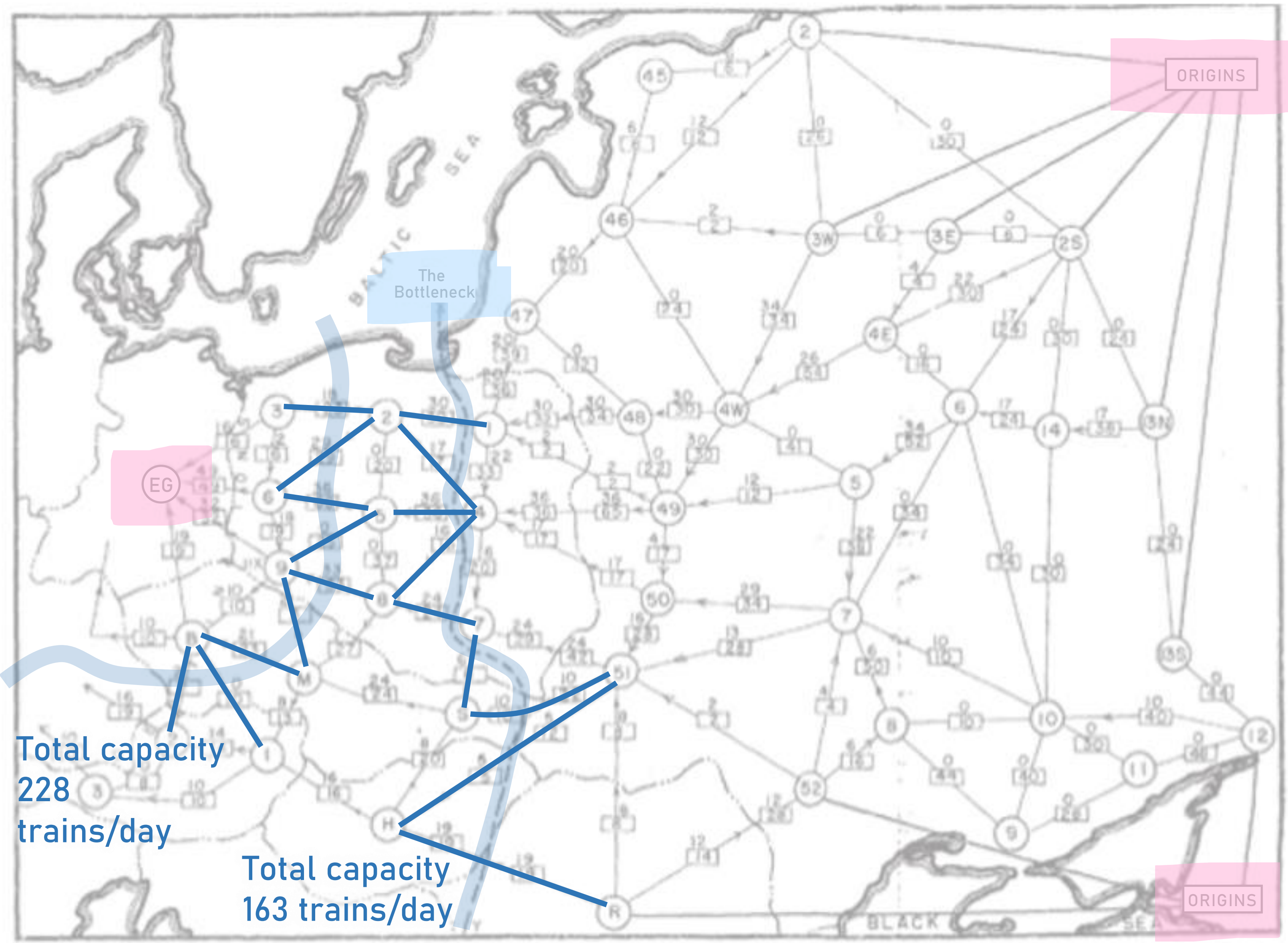


Fig. 7 — Traffic pattern: entire network available

Legend:

International boundary

Railway operating division

Capacity: 12 each way per day. Required flow of 9 per day toward destinations (in direction of arrow) with equivalent number of returning trains in opposite direction

All capacities in trains / 1000's of tons each way per day

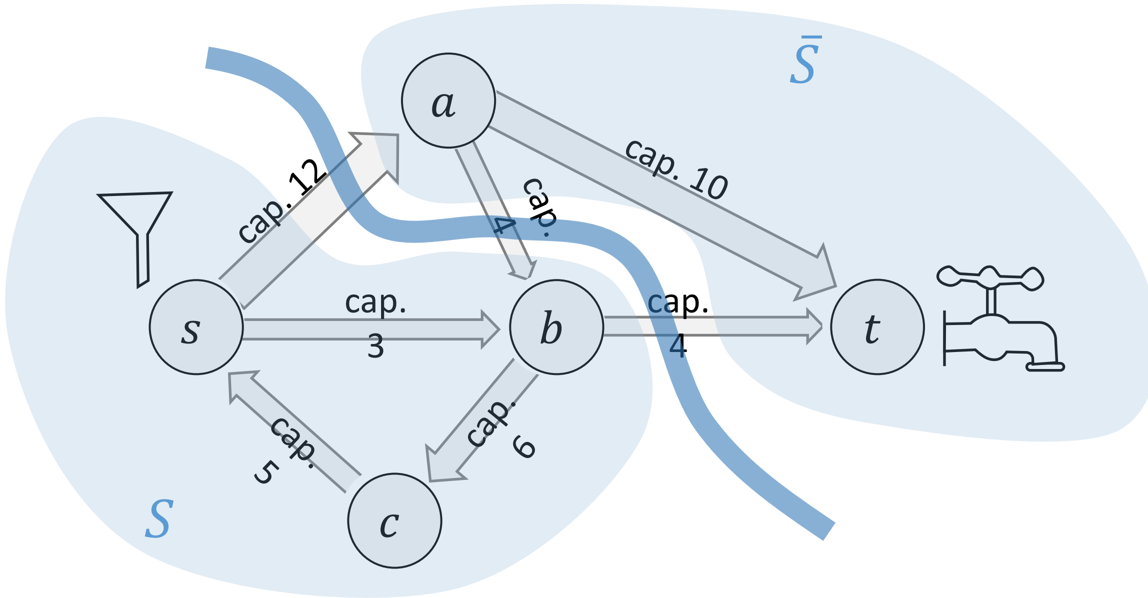
Origins: Divisions 2, 3W, 3E, 2S, 13N, 13S, 12, 52 (USSR), and Roumania

Destinations: Divisions 3, 6, 9 (Poland); B (Czechoslovakia); and 2, 3 (Austria)

Alternative destinations: Germany or East Germany

Note IIX of Division 9, Poland

A **cut** is a partition of the vertices into two sets, $V = S \cup \bar{S}$, with the source vertex $s \in S$ and the sink vertex $t \in \bar{S}$.



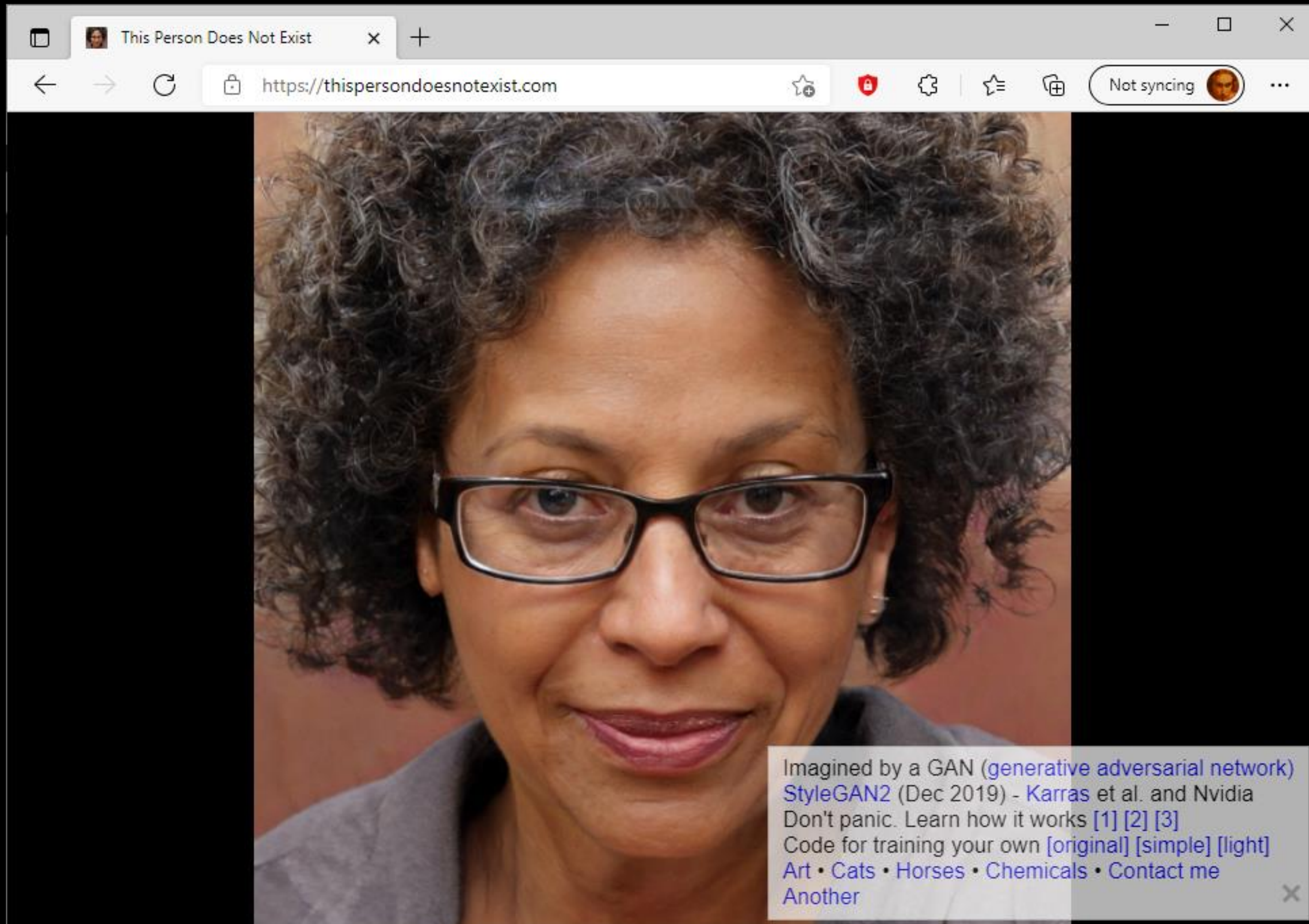
The **capacity** of the cut is

$$\text{capacity}(S, \bar{S}) = \sum_{\substack{u \in S,\, v \in \bar{S}: \\ u \to v}} c(u \to v)$$

MAX-FLOW MIN-CUT THEOREM

For any flow f and any cut $(S, \bar{S})$,

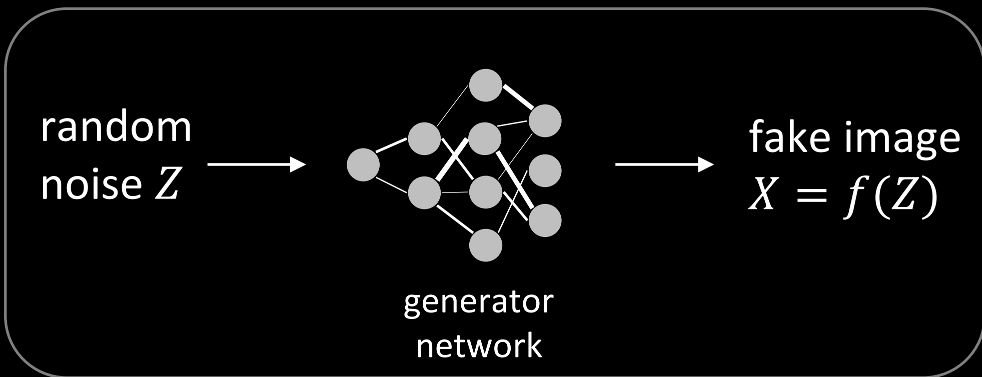
$$\text{value}(f) \le \text{capacity}(S, \bar{S})$$


This Person Does Not Exist
https://thispersondoesnotexist.com
Not syncing
Imagined by a GAN (generative adversarial network)
StyleGAN2 (Dec 2019) - Karras et al. and Nvidia
Don't panic. Learn how it works [1] [2] [3]
Code for training your own [original] [simple] [light]
Art • Cats • Horses • Chemicals • Contact me
Another

Given a dataset of images, how can we train a neural network to be able to generate realistic fakes?

Flickr-Faces-HQ Dataset (FFHQ)
https://github.com/NVlabs/ffhq-dataset

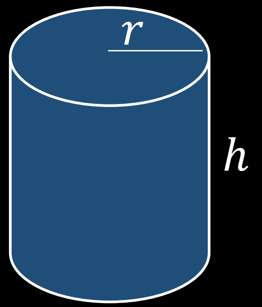
Given a dataset of images, how can we train a neural network to be able to generate realistic fakes?



This problem turns out to have close links to the max-flow min-cut theorem: they’re both about Lagrangian duality …

Maths for NST B lecture 17

A right-circular cylinder of radius r and height h has volume $\pi r^2 h$ and surface area $2\pi r^2 + 2\pi r h$. Given the surface area is A, find the largest possible volume.



$$\mathcal{L}(r,h;\lambda) = \pi r^2 h - \lambda(2\pi r^2 + 2\pi r h - A)$$

1. Write out the Lagrangian $\mathcal{L}$, as above
2. For a given λ, find $r \geq 0$ and $h \geq 0$ to maximize $\mathcal{L}(r,h;\lambda)$
3. Choose λ so that these r and h satisfy the constraint

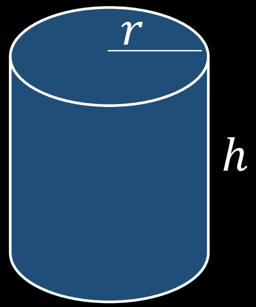
Step 2: $\frac{\partial \mathcal{L}}{\partial r} = 2\pi r h - \lambda(4\pi r + 2\pi h) = 0$, $\frac{\partial \mathcal{L}}{\partial h} = \pi r^2 - \lambda(2\pi r) = 0$ $\Rightarrow$ $h = 4\lambda$; ~~$r = 0$~~ or $r = 2\lambda$

Step 3: $2\pi r^2 + 2\pi r h = A \Rightarrow 8\pi\lambda^2 + 16\pi\lambda^2 = A \Rightarrow \lambda = \sqrt{\frac{A}{24\pi}}$

This gives $V = \pi r^2 h = 16\pi\lambda^3 = 16\pi\left(\frac{A}{24\pi}\right)^{3/2}$

Maths for NST B lecture 17

A right-circular cylinder of radius r and height h has volume $\pi r^2 h$ and surface area $2\pi r^2 + 2\pi r h$. Given the surface area is A, find the largest possible volume.



$$\mathcal{L}(r, h; \lambda) = \pi r^2 h - \lambda(2\pi r^2 + 2\pi r h - A)$$

For every λ, and for every (r, h) such that $\text{area}(r, h) = A$,

$$\begin{aligned}
\text{volume}(r, h) &= \pi r^2 h \\
&= \pi r^2 h - \lambda(\text{area}(r, h) - A) \\
&= \mathcal{L}(r, h; \lambda) \\
&\le \max_{r', h' \ge 0} \mathcal{L}(r', h'; \lambda) \\
&:= \text{cap}(\lambda)
\end{aligned}$$

Thus, for every λ,

$$\max_{r,h\,:\,\text{area}(r,h)=A} \text{volume}(r, h) \le \text{cap}(\lambda)$$

Thus,

$$\max_{r,h\,:\,\text{area}(r,h)=A} \text{volume}(r, h) \le \min_{\lambda} \text{cap}(\lambda)$$

This argument is called "weak Lagrangian duality"

Section 6.2

Given a weighted directed graph g with a source s and a sink t, find a flow from s to t with maximum possible value.

$$\text{maximize} \quad \text{val}(f) = \sum_{u: s\to u} f_{su} - \sum_{w: w\to s} f_{ws}$$

$$\text{over} \quad f \in \mathbb{R}^E, \quad 0 \le f_{uv} \le c_{uv} \text{ for all edges } u \to v$$

$$\text{such that} \quad \sum_{u: v\to u} f_{vu} - \sum_{w: w\to v} f_{wv} = 0 \text{ for all } v \in V \setminus \{s,t\}$$

The Lagrangian is

$$\mathcal{L}(f;\lambda) = \left(\sum_{u:s\to u} f_{su} - \sum_{w:w\to s} f_{ws}\right) - \sum_{v\ne s,t} \lambda_v \left(\sum_{u:v\to u} f_{vu} - \sum_{w:w\to v} f_{wv}\right)$$

Lagrangian weak duality says that for any flow f and for any λ,

$$\text{val}(f) \le \max_{f':0\le f'\le C} \mathcal{L}(f';\lambda)$$

The Lagrangian is

$$\mathcal{L}(f;\lambda) = \left(\sum_{u:s\to u} f_{su} - \sum_{w:w\to s} f_{ws}\right) - \sum_{v\neq s,t} \lambda_v \left(\sum_{u:v\to u} f_{vu} - \sum_{w:w\to v} f_{wv}\right)$$

Lagrangian weak duality says that for any flow f and for any λ,

$$\mathrm{val}(f) \leq \max_{f':0\leq f'\leq C} \mathcal{L}(f';\lambda)$$

$$\mathcal{L}(f';\lambda) = \sum_{v} \delta_v \left(\sum_{u:v\to u} f'_{vu} - \sum_{w:w\to v} f'_{wv}\right) \quad \text{where} \quad \delta_v = \begin{cases} 1 & \text{if } v=s \\ 0 & \text{if } v=t \\ \lambda_v & \text{otherwise} \end{cases}$$

$$= \sum_{v,u:v\to u} \delta_v f'_{vu} - \sum_{v,w:w\to v} \delta_w f'_{wv}$$

$$= \sum_{a,b:a\to b} \delta_a f'_{ab} - \sum_{b,a:a\to b} \delta_b f'_{ab}$$

$$= \sum_{a,b:a\to b} f'_{ab}(\delta_a - \delta_b)$$

This is maximized at $f'_{ab} = \begin{cases} c_{ab} & \text{if } \delta_a > \delta_b \\ 0 & \text{if } \delta_a < \delta_b \\ ? & \text{if } \delta_a = \delta_b \end{cases} = \begin{cases} c_{ab} & \text{if } a\in S, b\notin S \\ 0 & \text{if } a\notin S, b\in S \\ ? & \text{if } a,b \text{ on same side} \end{cases}$

Weak duality holds for any λ. Let's consider $\lambda_v = \begin{cases} 1 & \text{if } v\in S \\ 0 & \text{if } v\notin S \end{cases}$ for some cut $(S,\bar{S})$

For such a λ, $\max_{0\leq f'\leq C} \mathcal{L}(f';\lambda) = \sum_{a\in S,\, b\notin S} c_{ab} = \mathrm{capacity}(S,\bar{S})$

The max-flow min-cut theorem is an application of Lagrangian weak duality