Skyplane

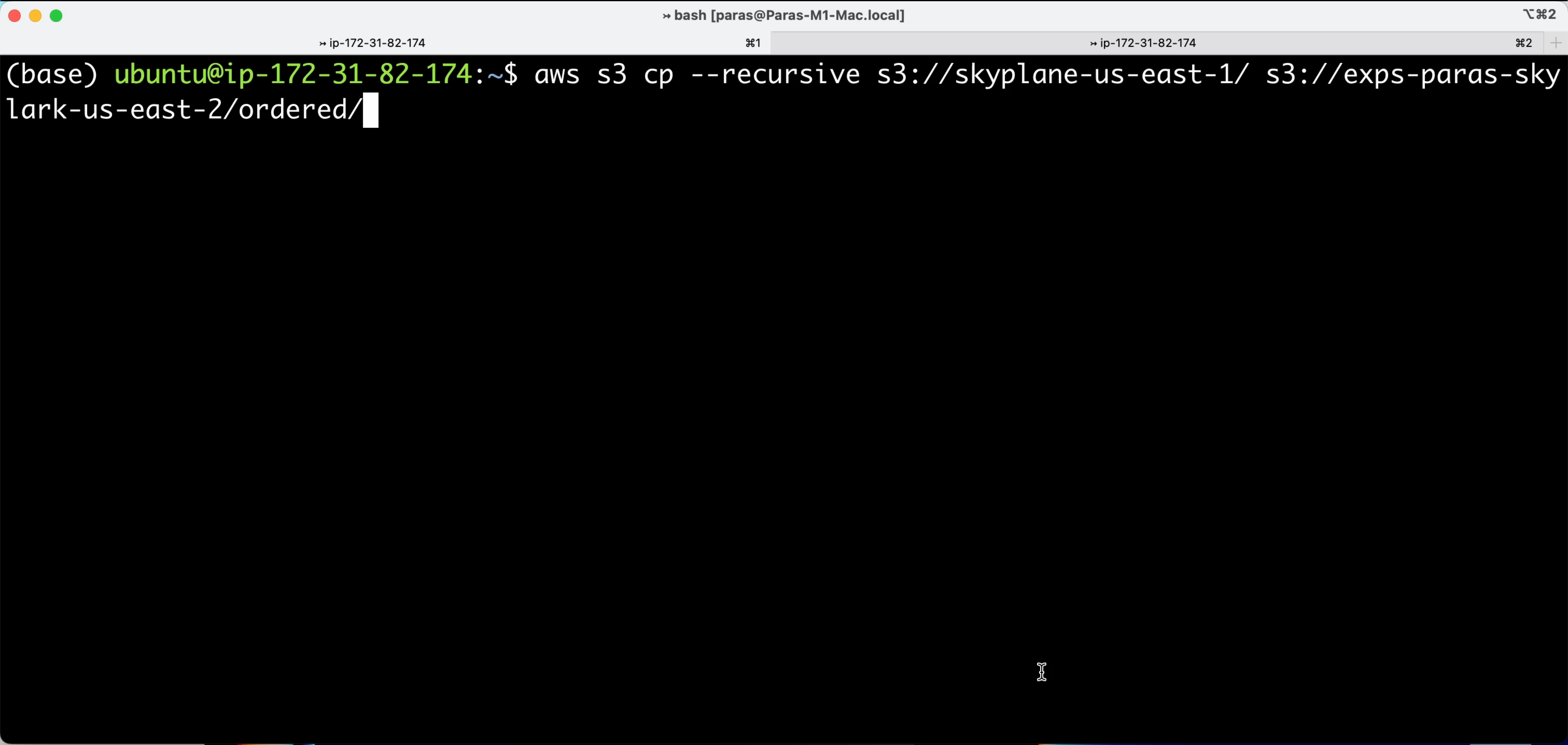
Skyplane: Optimizing Transfer Cost and Throughput Using Cloud-Aware Overlays

Paras Jain, Sam Kumar, Sarah Wooders, Shishir G. Patil, Joseph E. Gonzalez, and Ion Stoica

NSDI 2023 at Boston, MA

skyplane.org

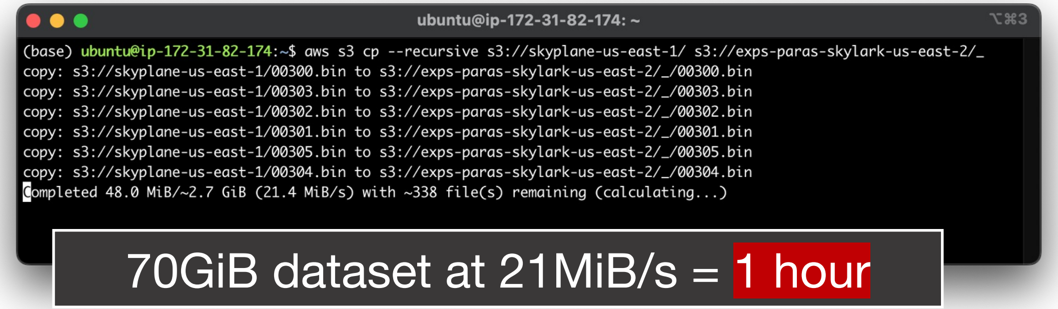
Working with data in the cloud is painful



```
(base) ubuntu@ip-172-31-82-174:~$ aws s3 cp --recursive s3://skyplane-us-east-1/ s3://exps-paras-sky
lark-us-east-2/ordered/
```

The problem of “**data gravity**”

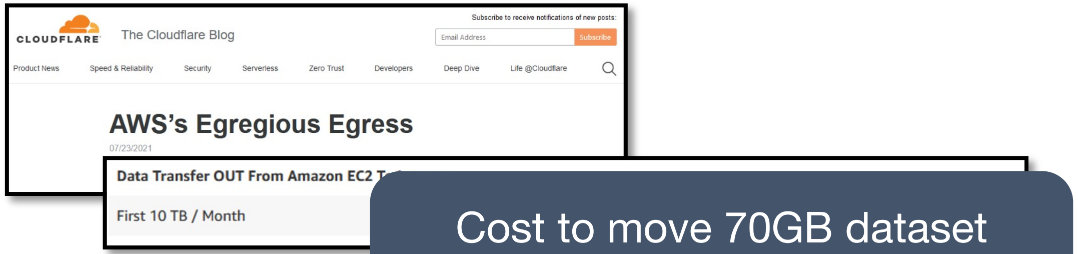
1. Slow transfers lock in data



```
(base) ubuntu@ip-172-31-82-174:~$ aws s3 cp --recursive s3://skyplane-us-east-1/ s3://exps-paras-skylark-us-east-2/_
copy: s3://skyplane-us-east-1/00300.bin to s3://exps-paras-skylark-us-east-2/_/00300.bin
copy: s3://skyplane-us-east-1/00303.bin to s3://exps-paras-skylark-us-east-2/_/00303.bin
copy: s3://skyplane-us-east-1/00302.bin to s3://exps-paras-skylark-us-east-2/_/00302.bin
copy: s3://skyplane-us-east-1/00301.bin to s3://exps-paras-skylark-us-east-2/_/00301.bin
copy: s3://skyplane-us-east-1/00305.bin to s3://exps-paras-skylark-us-east-2/_/00305.bin
copy: s3://skyplane-us-east-1/00304.bin to s3://exps-paras-skylark-us-east-2/_/00304.bin
Completed 48.0 MiB/~2.7 GiB (21.4 MiB/s) with ~338 file(s) remaining (calculating...)
```

70GiB dataset at 21MiB/s = 1 hour

2. High egress fees = $$$



The Cloudflare Blog

AWS’s Egregious Egress

07/23/2021

Data Transfer OUT From Amazon EC2 T

First 10 TB / Month

Cost to move 70GB dataset

= running **34 instances** (m5.xlarge)

skyplane.org

The problem of “**data gravity**”

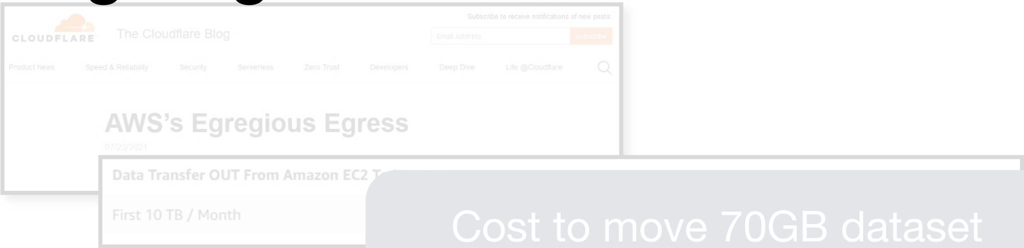
How to solve data gravity?

1) Slow transfer speeds
2) High egress fees

```
(base) ubuntu@ip-172-31-82-174:~$ aws s3 cp --recursive s3://skyplane-us-east-1/ s3://exps-paras-skylark-us-east-2/_
```

70GiB dataset at 2 MiB/s = 1 hour

2. High egress fees = $$$



The Cloudflare Blog

Subscribe to receive notifications of new posts:

Email Address

Subscribe

Product News | Speed & Reliability | Security | Serverless | Zero Trust | Developers | Deep Dive | Life @Cloudflare

AWS’s Egregious Egress

07/23/2021

Data Transfer OUT From Amazon EC2 T

First 10 TB / Month

Cost to move 70GB dataset
= running **34 instances** (m5.xlarge)

What is Skyplane?

Problem: Managing data across regions and across clouds is **slow** and **expensive**

What is Skyplane?

Problem: Managing data across regions and across clouds is **slow** and **expensive**

Skyplane is a system for fast, low-cost transfers between object stores.

```
skyplane cp {s3,gs,az}://... {s3,gs,az}://...
```

What is Skyplane?

Problem: Managing data across regions and across clouds is **slow** and **expensive**

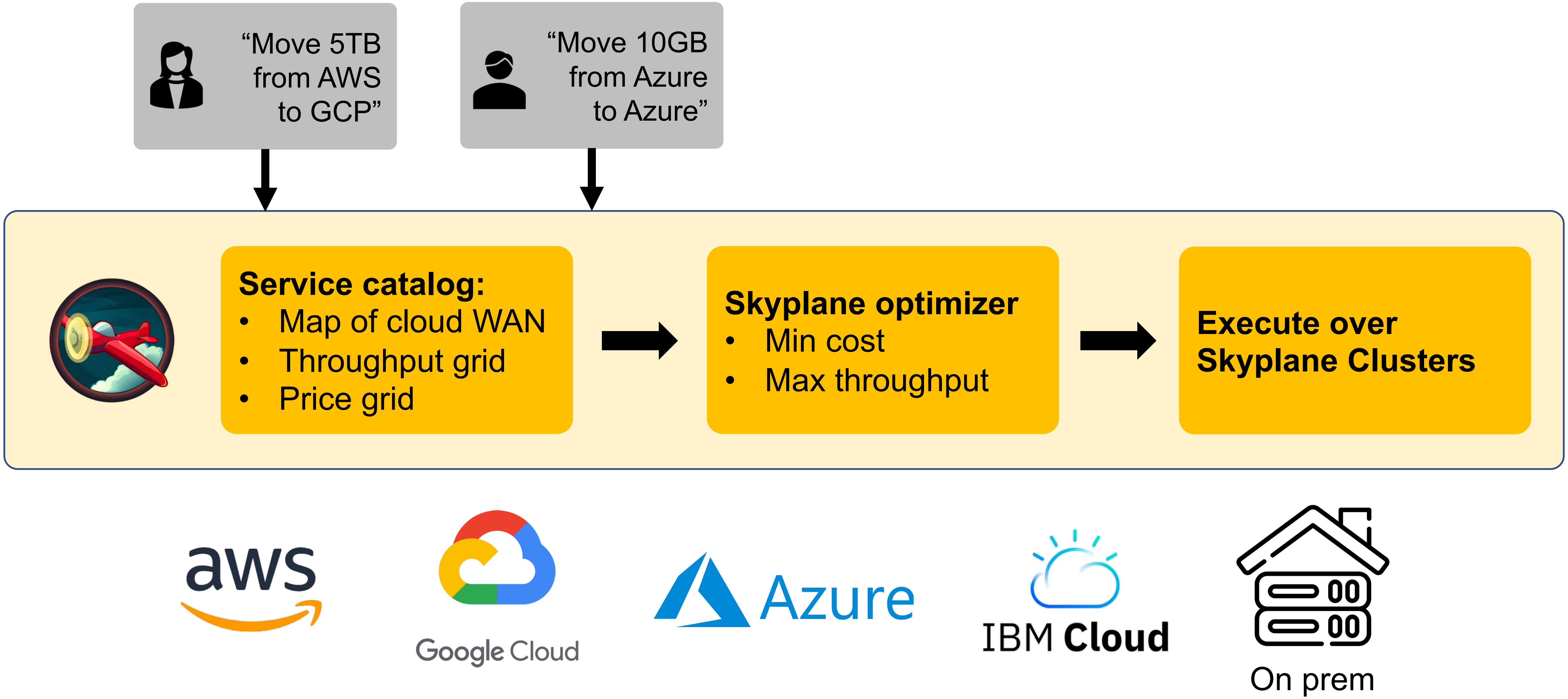
Skyplane is a system for fast, low-cost transfers between object stores.

```
skyplane cp {s3,gs,az}://... {s3,gs,az}://...
```

How does it work?

1. **Profiling:** Probe cloud network throughput
2. **Planning:** Centralized LP planner finds optimal transfer path
3. **Execution:** Provision ephemeral gateway VMs from plan

Sky computing: Intercloud Broker for data transfer



"Move 5TB from AWS to GCP"

"Move 10GB from Azure to Azure"

Service catalog:

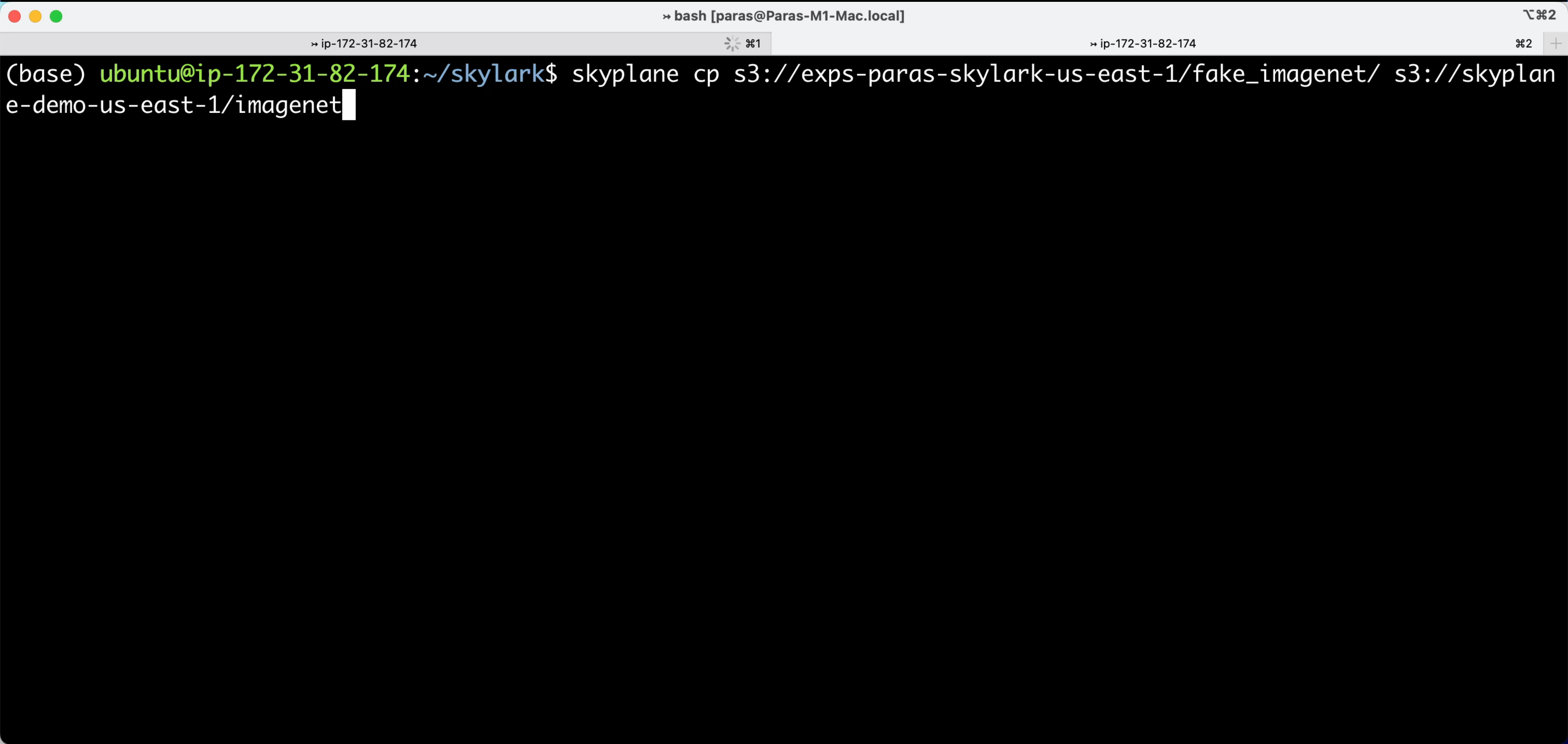
- Map of cloud WAN
- Throughput grid
- Price grid

Skyplane optimizer

- Min cost
- Max throughput

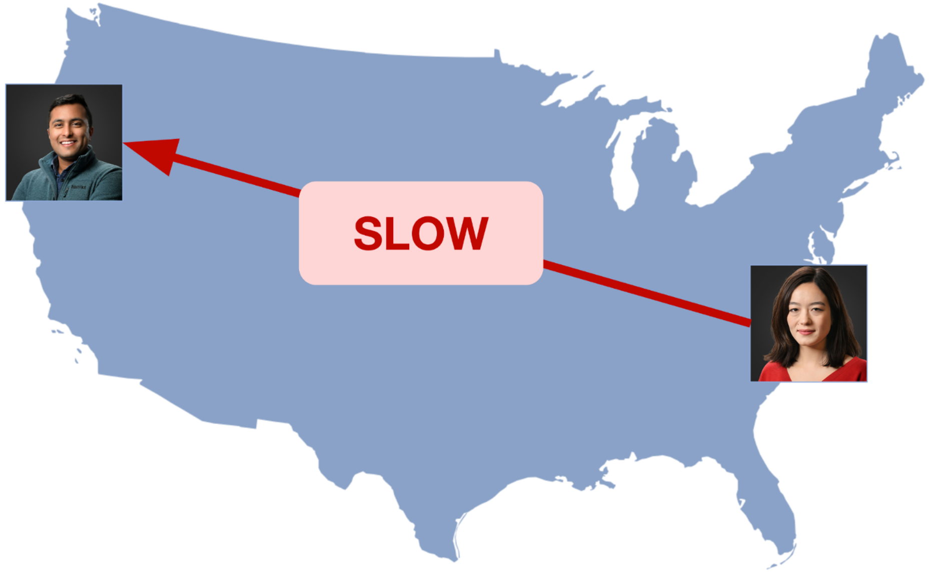
Execute over Skyplane Clusters

This paper: high speed, low cost data transfers with the Skyplane transfer broker



```
(base) ubuntu@ip-172-31-82-174:~/skylark$ skyplane cp s3://exps-paras-skylark-us-east-1/fake_imagenet/ s3://skyplan
e-demo-us-east-1/imagenet
```

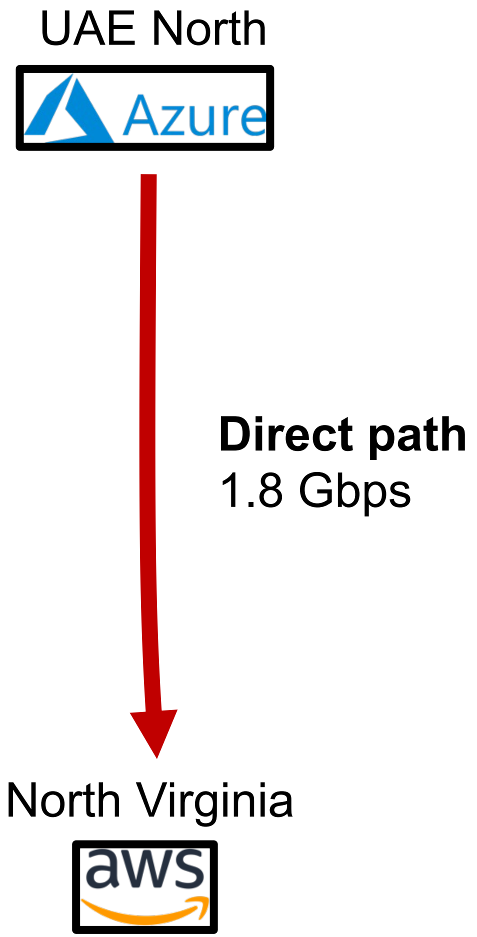
Direct internet path between clouds are often slow



Reasons for slow transfers

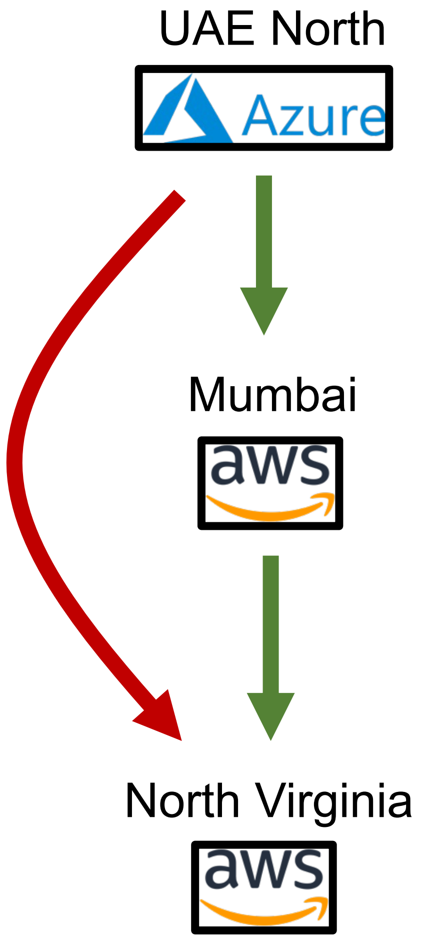
1. Congestion along direct path
2. Poor peering between providers
3. Packet loss from the physical layer
4. (surprising) Throttling from cloud providers

Insight #1: overlay routing to circumvent slow links



Direct path
1.8 Gbps

Insight #1: overlay routing to circumvent slow links



Direct path
1.8 Gbps

Indirect path
4.6Gbps

Insight #1: overlay routing to circumvent slow links

Overlay routing is
a classic method

RON [SOSP 2001]
Chord [SIGCOMM 2001]
Bullet [SOSP 2003]
Akamai [SIGOPS 2010]
Baidu BDS [EuroSys 2018]
and countless others…

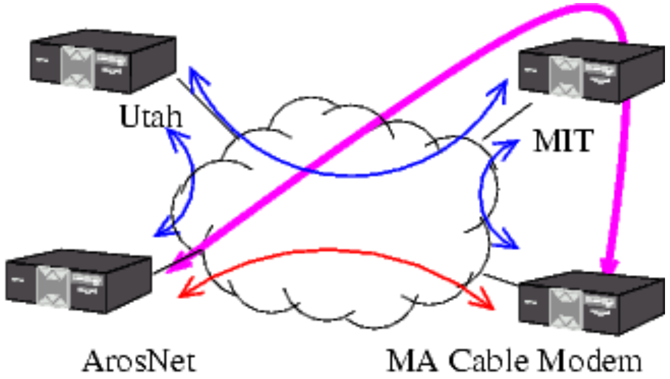


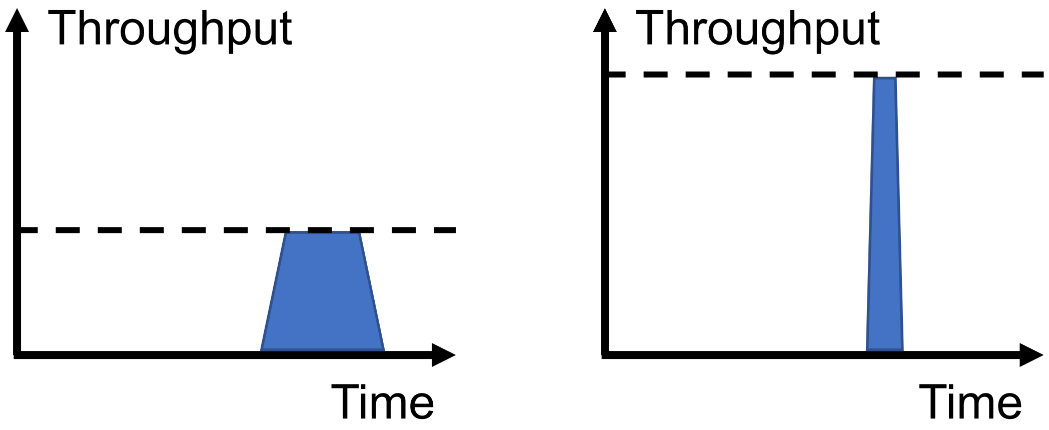
Figure from RON

Insight #1: classic overlay routing is not designed in the cloud

Overlay routing is a classic method

RON [SOSP 2001]
Chord [SIGCOMM 2001]
Bullet [SOSP 2003]
Akamai [SIGOPS 2010]
Baidu BDS [EuroSys 2018]
and countless others…

Novel problem space: network + VM pricing



In the cloud, you pay for the area under the curve
1Mbps for 40 days = 1Gbps for 1 hour

Insight #1: classic overlay routing is not designed in the cloud

Overlay routing is a classic method

RON [SOSP 2001]
Chord [SIGCOMM 2001]
Bullet [SOSP 2003]
Akamai [SIGOPS 2010]
Baidu BDS [EuroSys 2018]
and countless others…

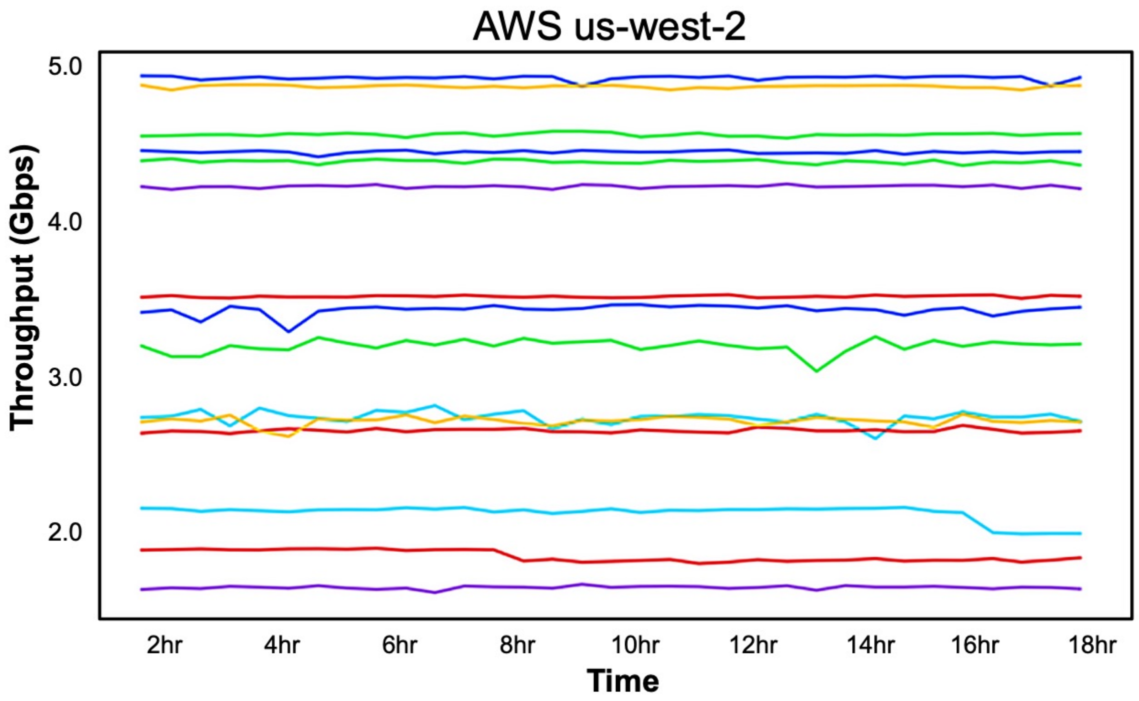
Novel problem space: network + VM pricing

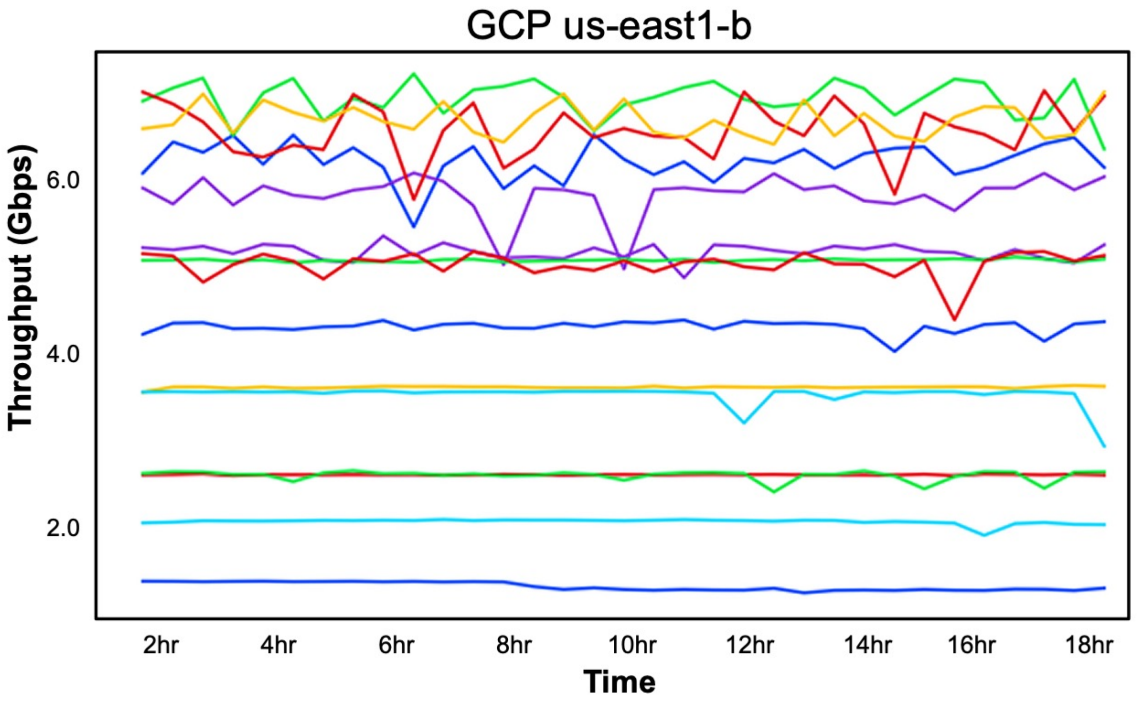
- Classic assumption: networks are free or priced by throughput
- Cloud is priced per unit volume ($ per GB transferred)

Novel solution space: elasticity

- Classic assumption: fixed overlay locations each without parallelism
- Cloud supports elasticity in location and # of VMs

Insight #1: Applying optimization to search the cost-throughput tradeoff space

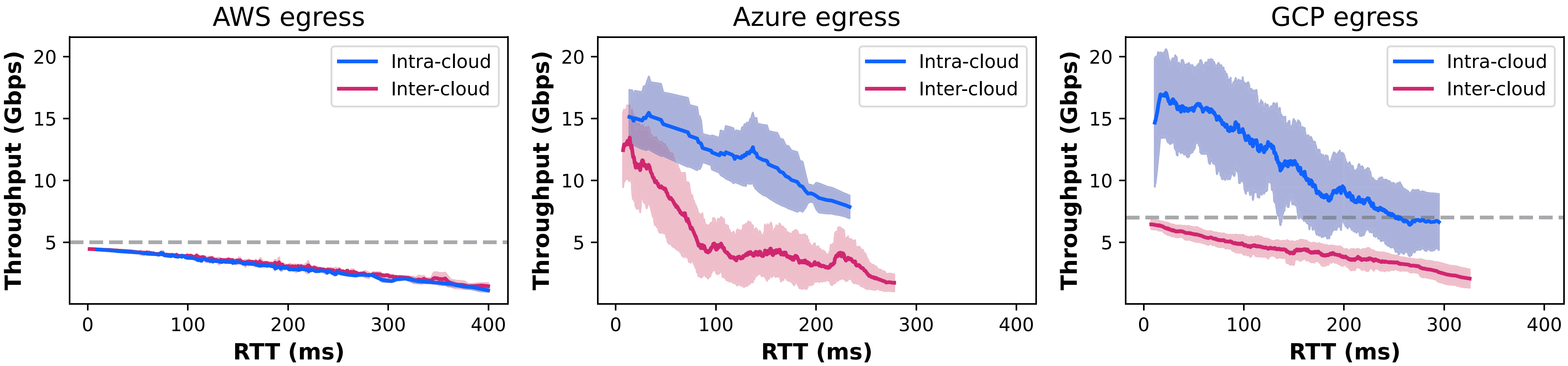




Egress speeds in the cloud are stable over a 24 hour period

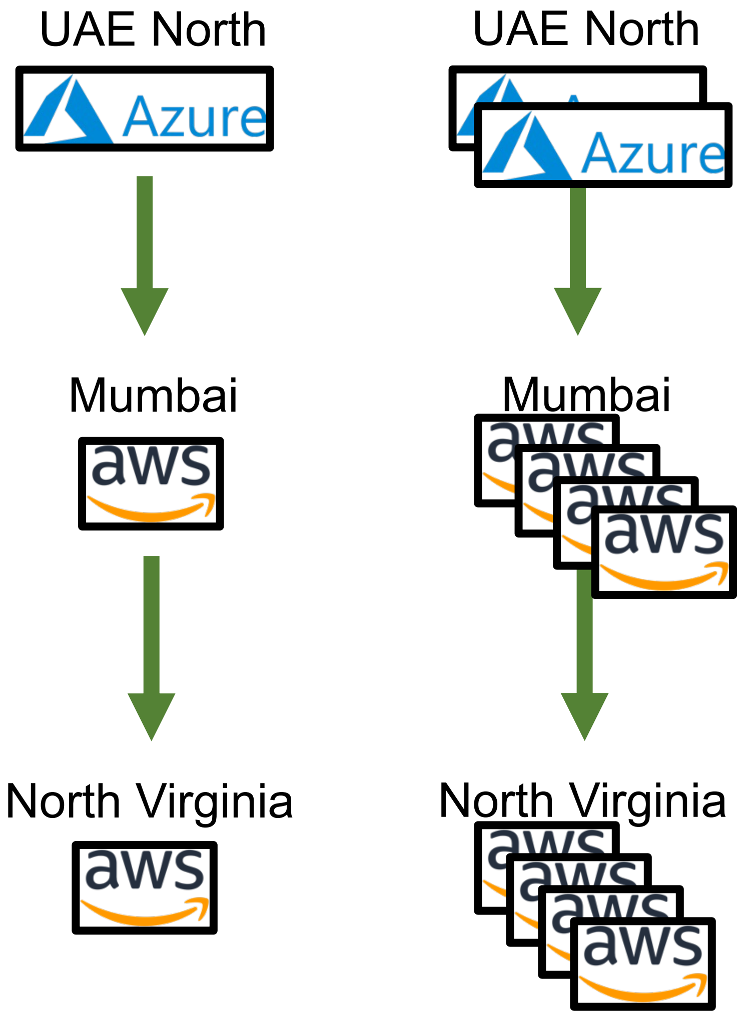
→ Centralized planning is feasible

Insight #2: parallel VMs per region to avoid provider throttling



Clouds throttle egress speeds!

Insight #2: parallel VMs per region to avoid provider throttling



Before: throttled to 5Gbps

After: transfer at up to 20Gbps

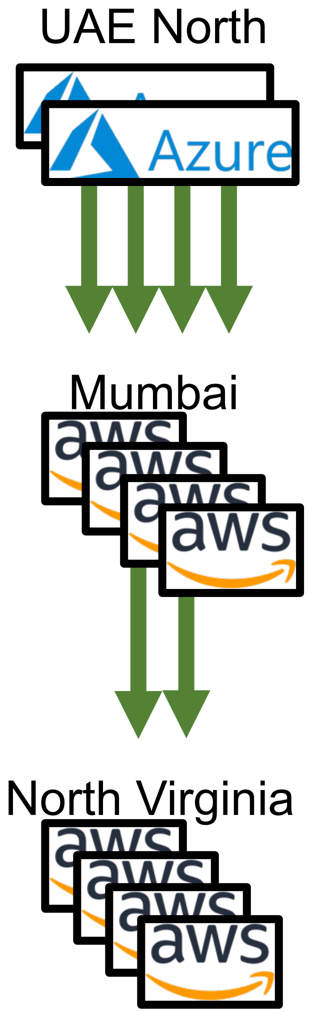
Overlay routing

Longer indirect paths are worthwhile for slow links

of VMs per region

Access throughput beyond NIC, AWS and GCP throttle egress

Insight #3: parallel TCP connections to improve goodput



Overlay routing

Longer indirect paths are worthwhile for slow links

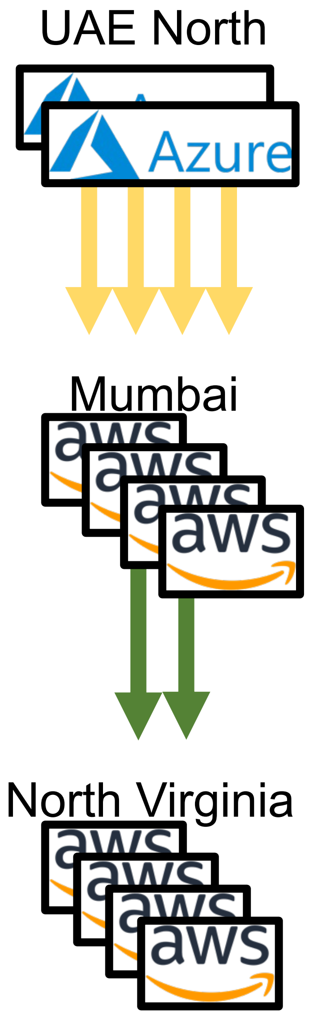
of VMs per region

Access throughput beyond NIC, AWS and GCP throttle egress

# of parallel TCP connections

Inspired by GridFTP, but must consider VM and NIC limits

Insight #4: cut cost with compression + network tiers



Overlay routing

Longer indirect paths are worthwhile for slow links

of VMs per region

Access throughput beyond NIC, AWS and GCP throttle egress

# of parallel TCP connections

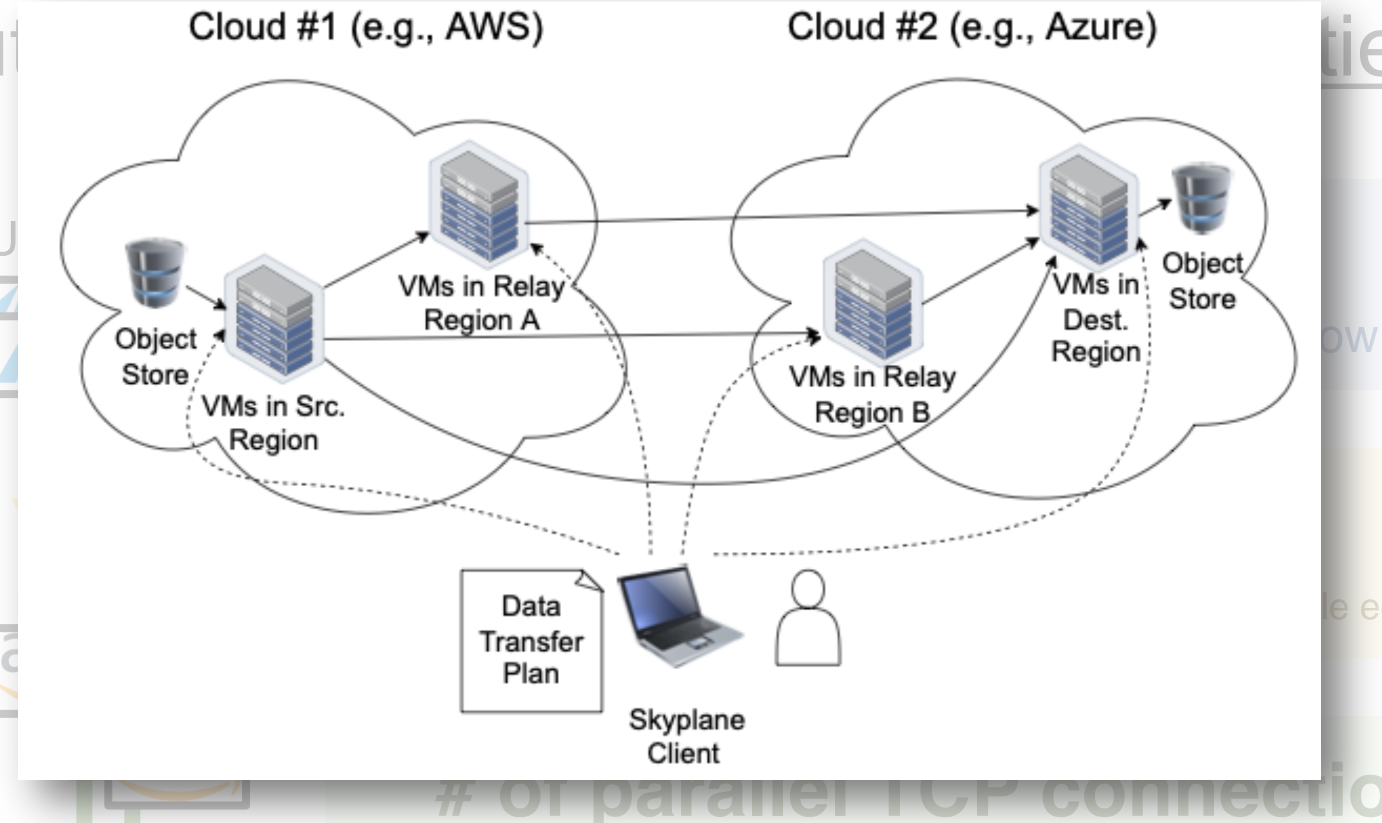
Inspired by GridFTP, but must consider VM and NIC limits

Network tiering + compression

Hot potato routing up to 40% cheaper than cold potato

Insight #4: cu tiers

U



ow links

le egress

# of parallel TCP connections

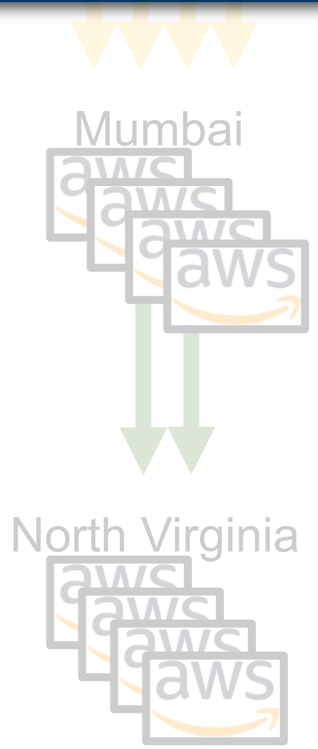
No cooperation required from clouds!

Skyplane only uses public APIs + runs in your cloud VPC

Insight #4: cut cost with compression + network tiers

Open source project!

```
$ pip install skyplane[aws]
```



# of VMs per region

AWS and GCP throttle egress

P connections

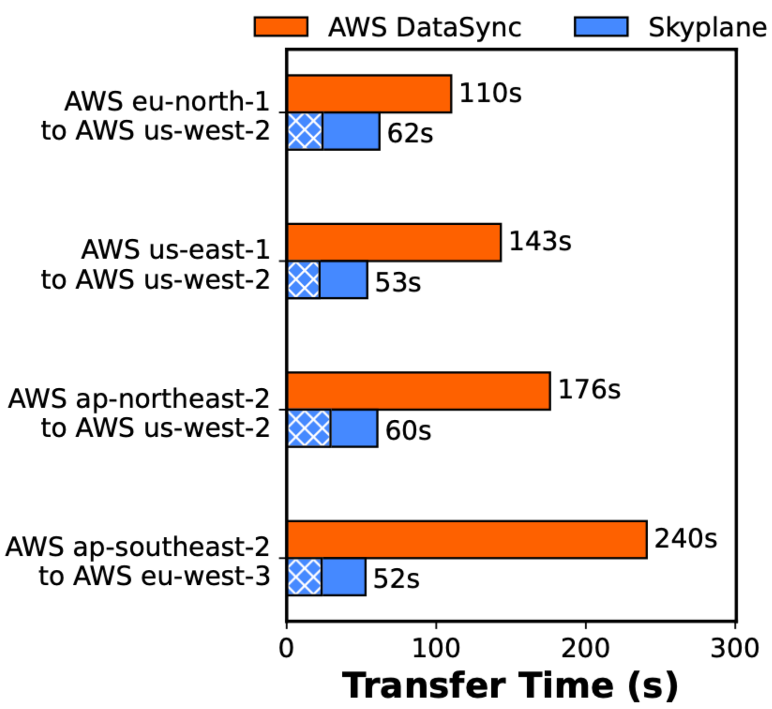
consider VM and NIC limits

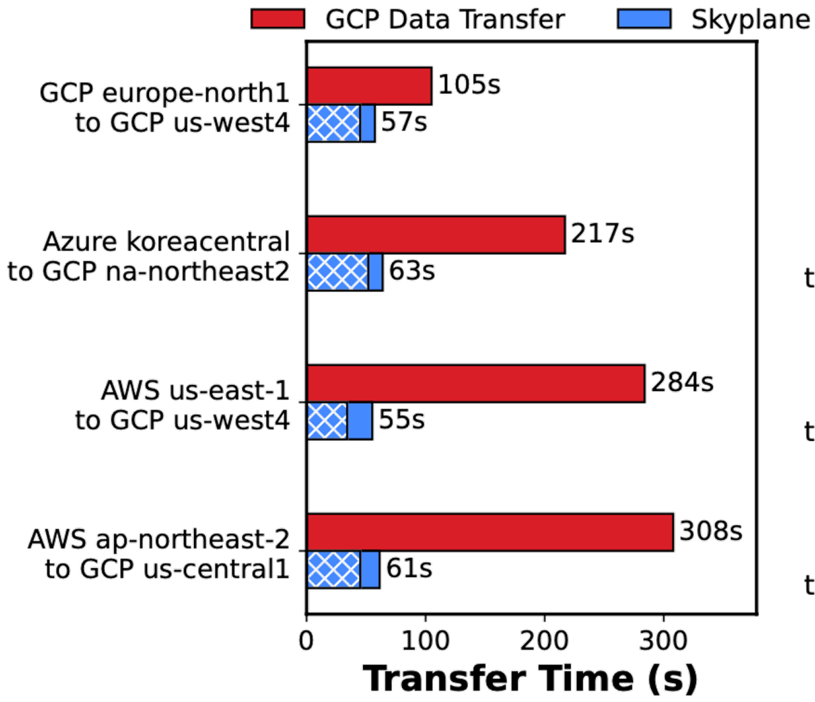
+ compression

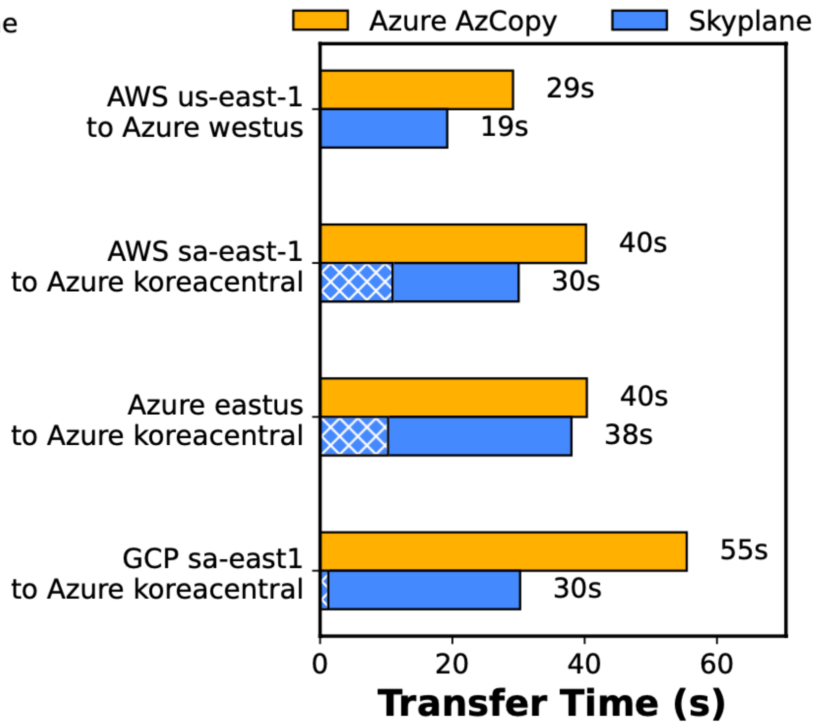
cheaper than cold potato

skyplane.org

Evaluation: End-to-end comparison against cloud providers







Versus AWS Datasync:

- Up to 4.6x faster for AWS-AWS
- DataSync did not support intercloud

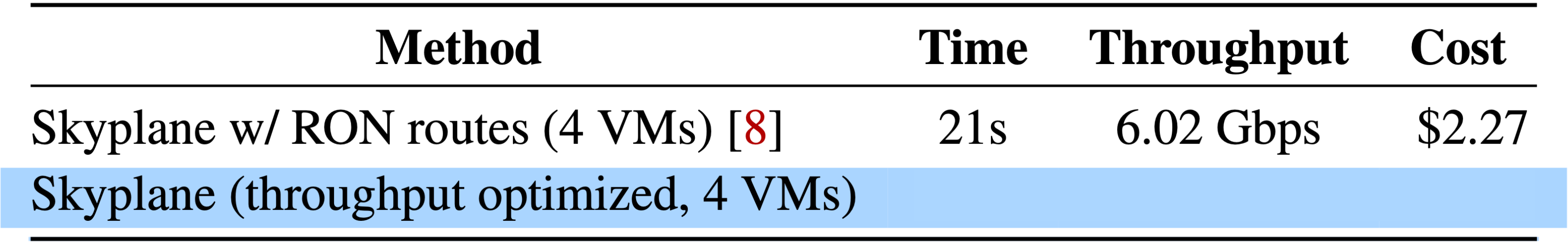
Versus GCP Data Transfer:

- Up to 1.8x faster for GCP-GCP
- Up to 5.0x faster for AWS to GCP
- GCP egress not supported

Versus Azure AzCopy

- Similar speeds for Azure-Azure
- Up to 1.8x faster for GCP to Azure
- Why? AzCopy leverages compute inside Azure Blob

Evaluation: Comparison to Resilient Overlay Networks

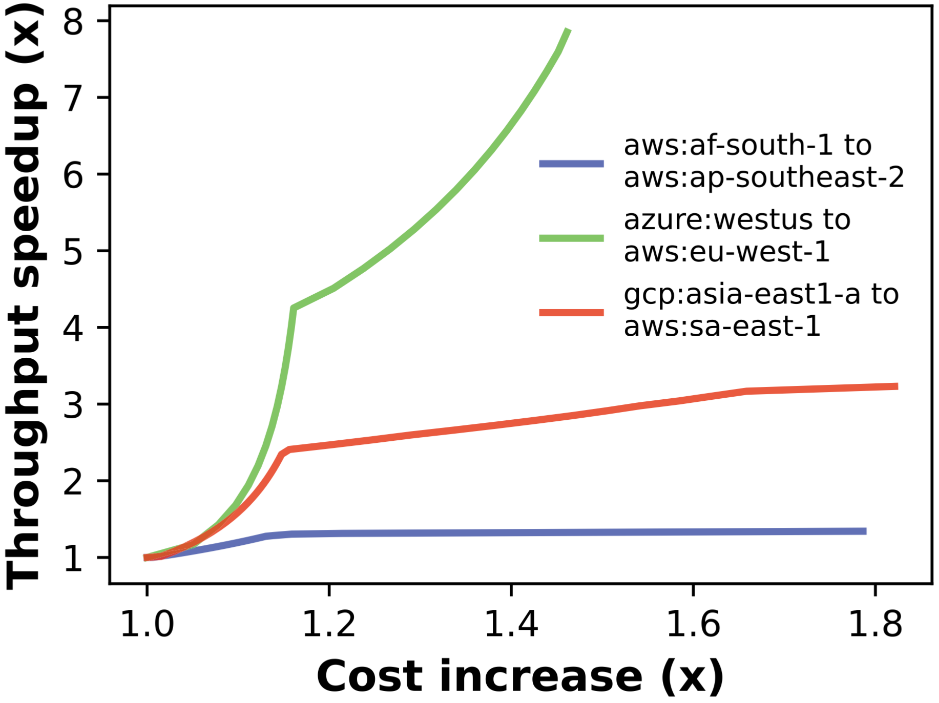


Method	Time	Throughput	Cost
Skyplane w/ RON routes (4 VMs) [8]	21s	6.02 Gbps	$2.27
Skyplane (throughput optimized, 4 VMs)			

To compare with RON, we implemented the route from RON's optimizer in Skyplane

- 16GB transfer from Azure `East US` to AWS `ap-northeast-1`
- Compression + tiering disabled for these experiments
- **Result:** 1.3x speedup at 30% lower cost than RON

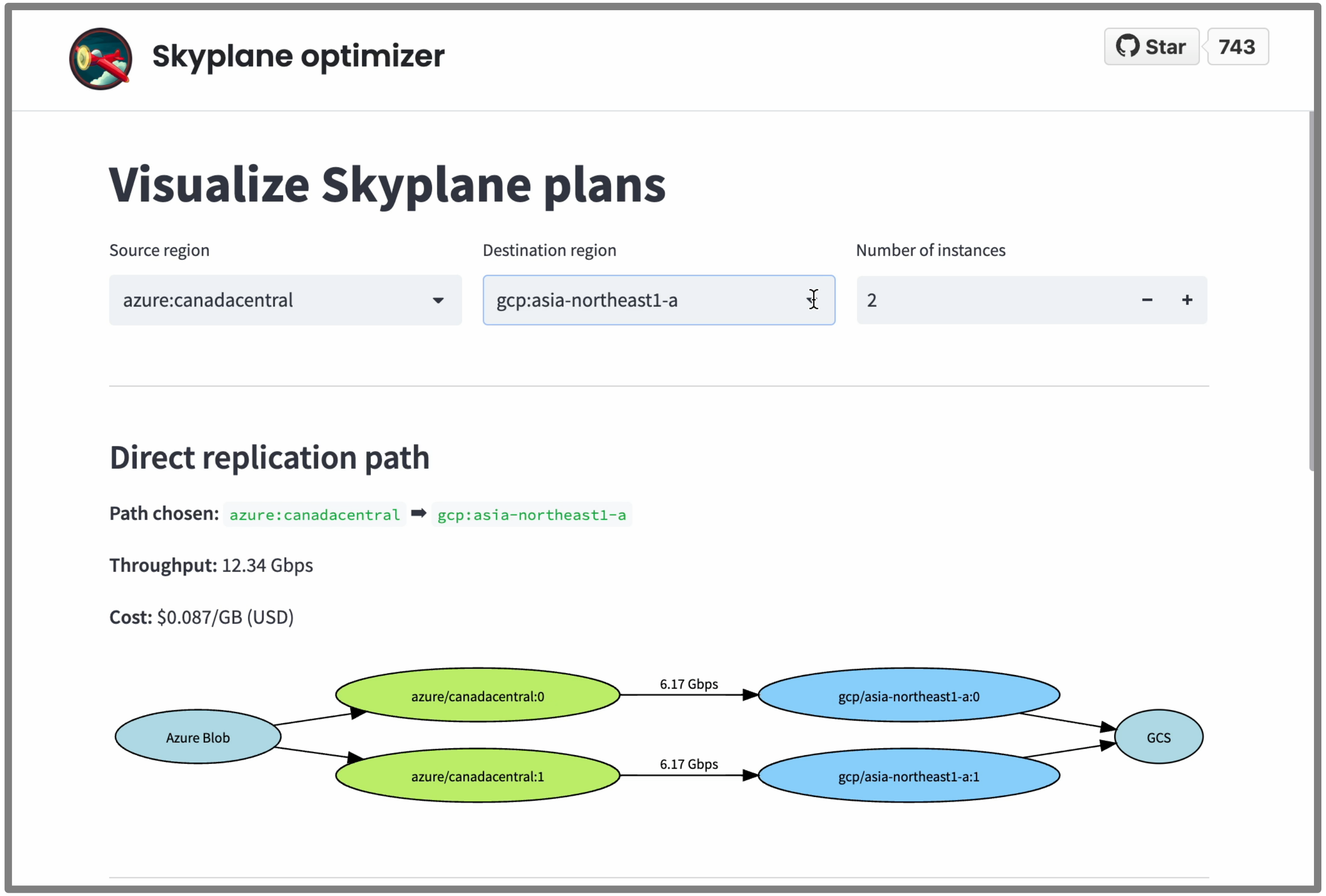
Evaluation: Visualizing the cost-throughput space



Skyplane can achieve substantial improvements in transfer speeds with minimal cost increases

4x throughput improvement for a 20% premium

Try out Skyplane’s optimizer



Skyplane optimizer

Visualize Skyplane plans

Source region: azure:canadacentral

Destination region: gcp:asia-northeast1-a

Number of instances: 2

Direct replication path

Path chosen: `azure:canadacentral` ➡ `gcp:asia-northeast1-a`

Throughput: 12.34 Gbps

Cost: $0.087/GB (USD)

https://optimizer.skyplane.org/

Open-source adoption

skyplane-project / skyplane Public

Approaching ½ PiB transferred!

Apache 2.0 licensed project
https://github.com/skyplane-project/skyplane

Gridware · Berkeley UNIVERSITY OF CALIFORNIA · BROAD INSTITUTE · NASA EARTHDATA Powered by EOSDIS

VOXEL · Stanford University · ATLASSIAN · kaliberai™

and many more users + contributors!

Skyplane team

A big team effort at UC Berkeley Sky computing

Shu Liu

Sam Kumar

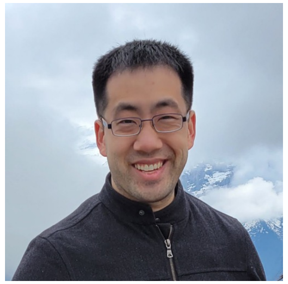
Sarah Wooders

Paras Jain

Shishir Patil

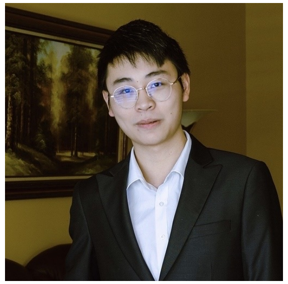
Ion Stoica

Joey Gonzalez

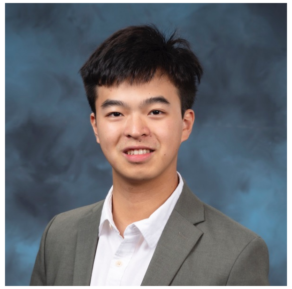

Vincent Liu

Daniel Kang

Asim Biswal


Jason Ding

Anton Zabreyko


Xuting Liu

Hailey Jang

Simon Mo

Skyplane

Optimizing Transfer Cost and Throughput Using Cloud-Aware Overlays

Problem: cross-region and cross-cloud transfers are slow and expensive

Skyplane accelerates cloud transfers while reducing egress costs

Open-source tool – please share feedback, use cases or collaborations!

```
$ pip install skyplane[aws,azure,gcp]
$ skyplane init
$ skyplane cp -r s3://… gcs://…
```

skyplane.org

parasj@berkeley.edu