nsdi '23

StarryNet

Empowering Researchers to Evaluate Futuristic Integrated Space and Terrestrial Networks

Zeqi Lai, Hewu Li, Yangtao Deng, Qian Wu, Jun Liu, Yuanjie Li,
Jihao Li, Lixin Liu, Weisen Liu, Jianping Wu

Presenter: Yangtao Deng

清華大學
Tsinghua University

CONTENTS

The Future is Up in the Sky

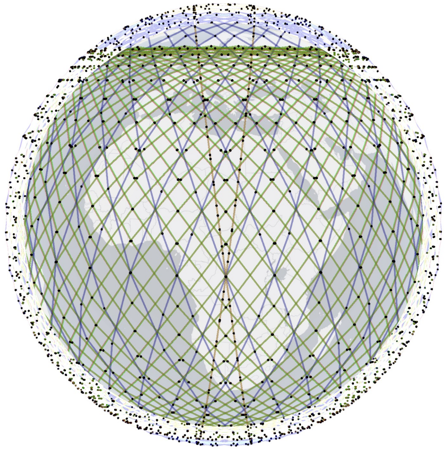
Satellite Internet constellations are under heavy development

The Future is Up in the Sky

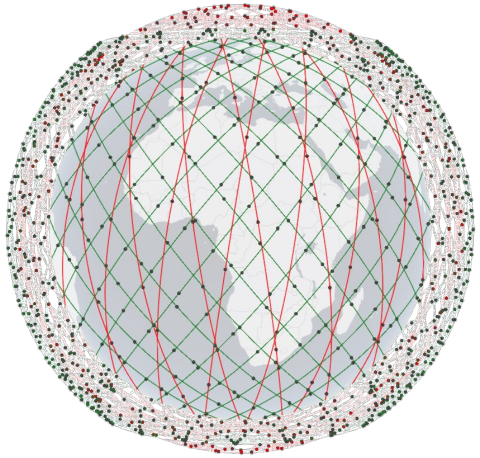
Tsinghua University

Satellite Internet constellations are under heavy development

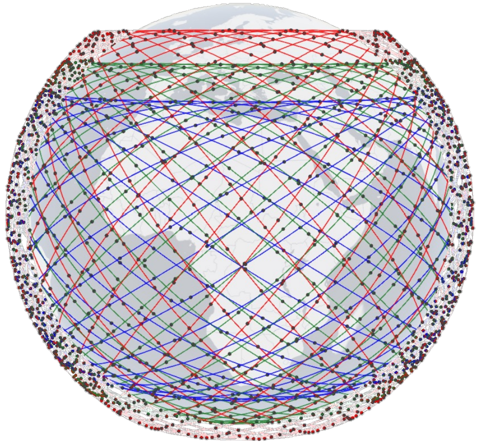
Thousands of broadband satellites in low earth orbit (LEO)



4408 satellites in 5 shells



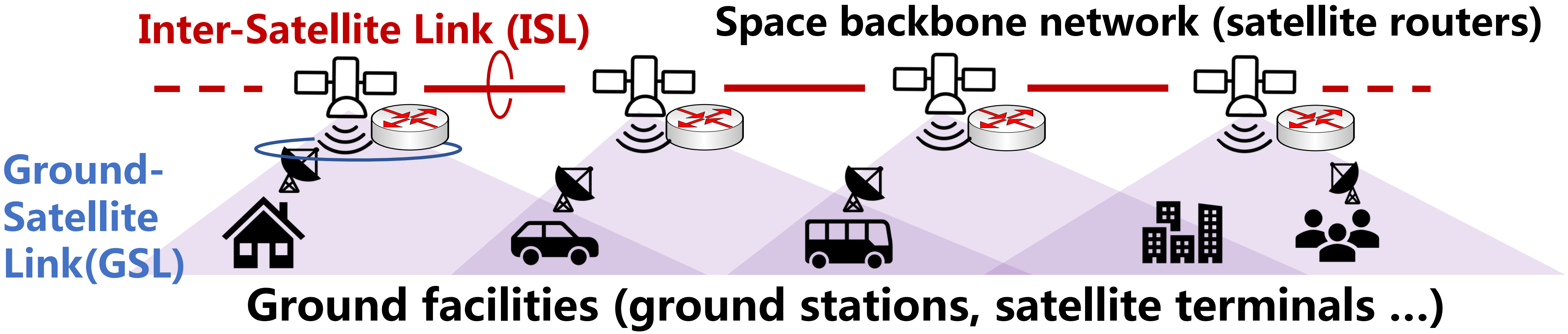
1671 satellites in 2 shells



3236 satellites in 3 shells

Integrated Space and Terrestrial Network

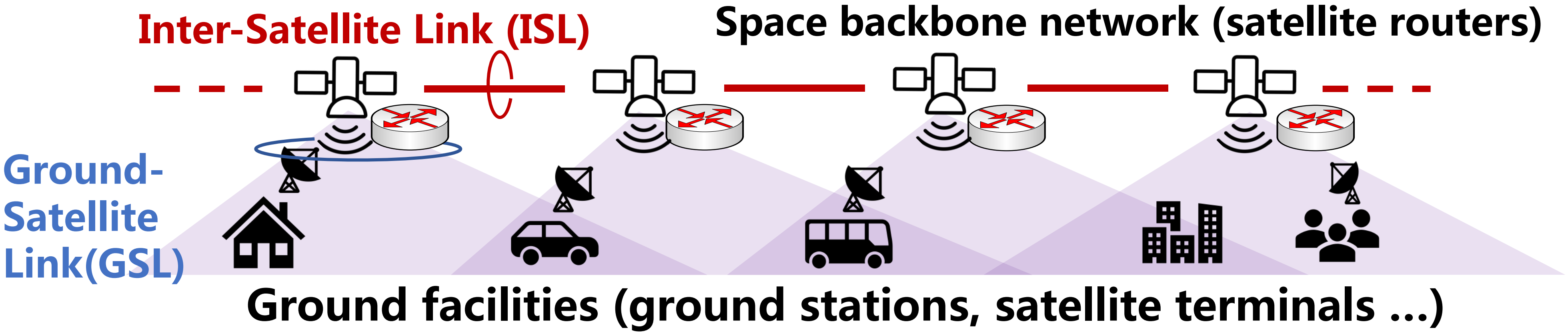
Integrating LEO satellites with existing terrestrial Internet (ISTN)



Integrated **S**pace and **T**errestrial **N**etwork

清華大學 Tsinghua University

Integrating LEO satellites with existing terrestrial Internet (ISTN)



Remote Service

Rural Education

Airplane

Global IoT

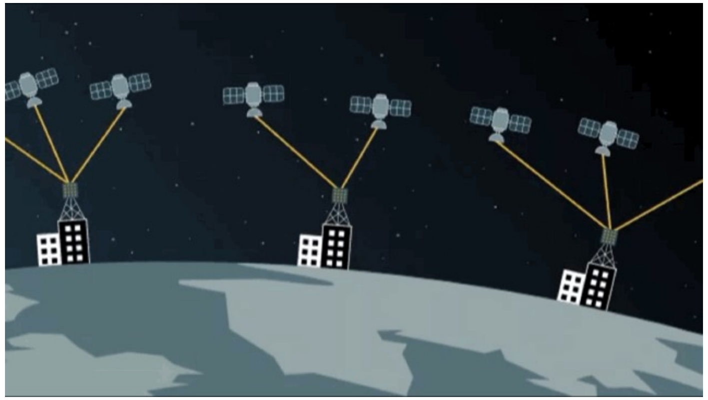
Maritime

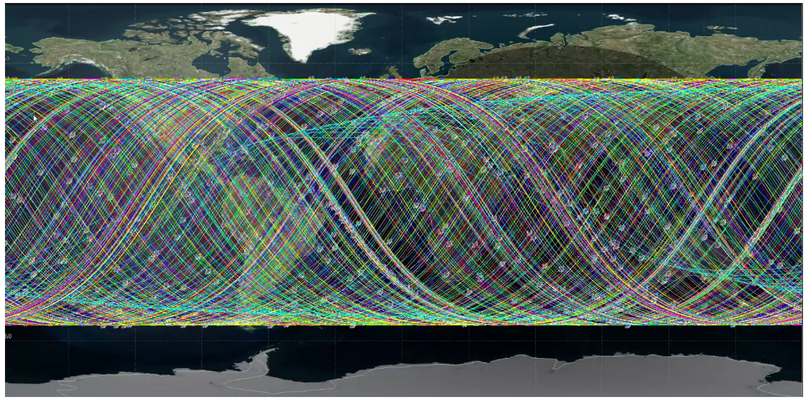
Provide pervasive, low-latency, high-bandwidth Internet service

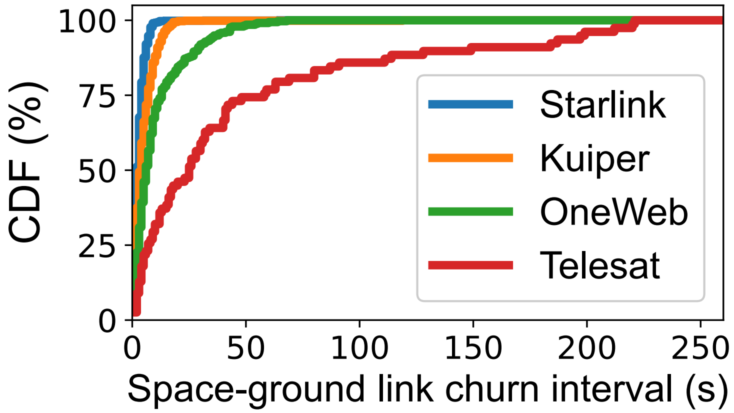
Unique Characteristics of ISTN

Satellites move at a high velocity in the outer space

resulting in high LEO dynamics and NEW challenges on the networking stack





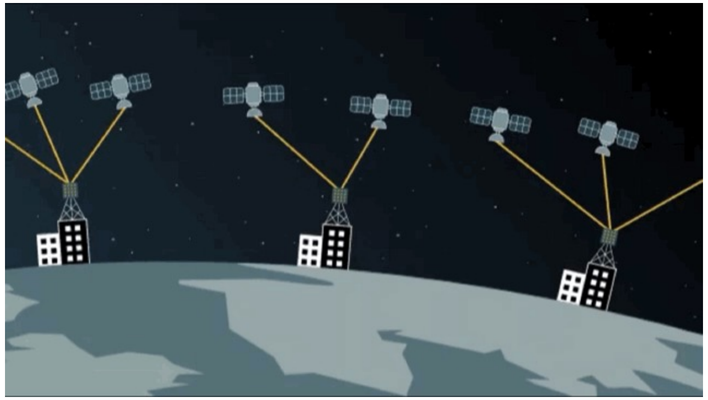


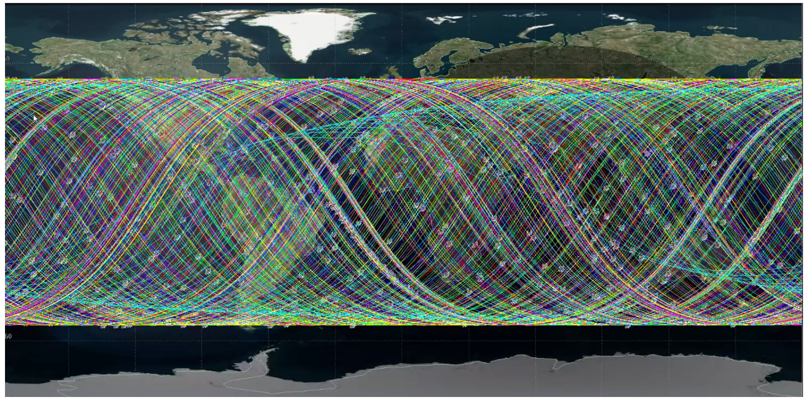
[*]

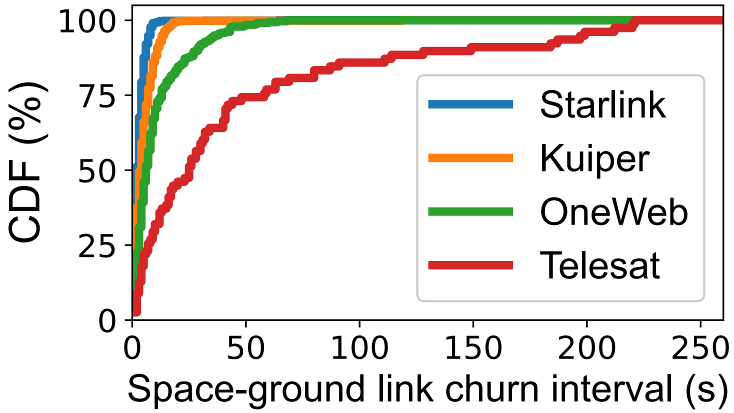
[*] "Internet in Space" for Terrestrial Users via Cyber-Physical Convergence, HotNets'21

Unique Characteristics of ISTN

Satellites move at a high velocity in the outer space
resulting in high LEO dynamics and NEW challenges on the networking stack







[*]

Researcher may propose NEW networking technologies to tackle those challenges (e.g. a new ground-satellite integration scheme).

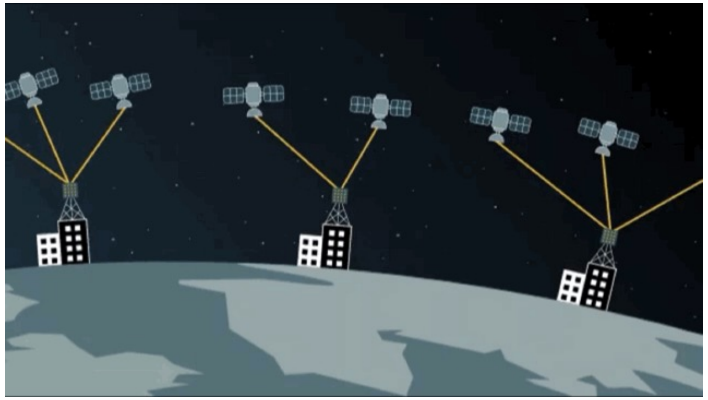
[*] "Internet in Space" for Terrestrial Users via Cyber-Physical Convergence, HotNets’21

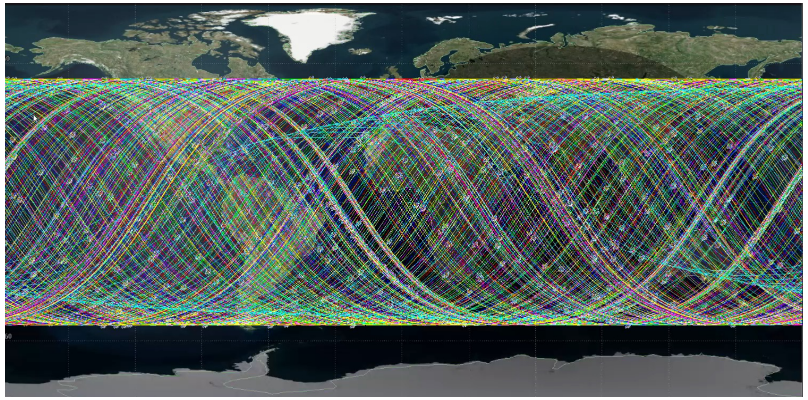
Unique Characteristics of ISTN

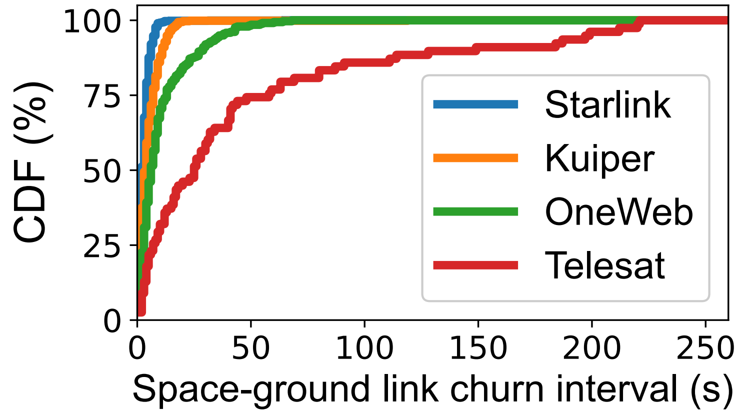
Tsinghua University

Satellites move at a high velocity in the outer space

resulting in high LEO dynamics and NEW challenges on the networking stack







[*]

Researcher may propose NEW networking technologies to tackle those challenges (e.g. a new ground-satellite integration scheme).

How can researchers build an experimental network environment (ENE) to test, evaluate and understand their new ideas?

[*] "Internet in Space" for Terrestrial Users via Cyber-Physical Convergence, HotNets'21

ENE Requirements for ISTN Experiments

Tsinghua University

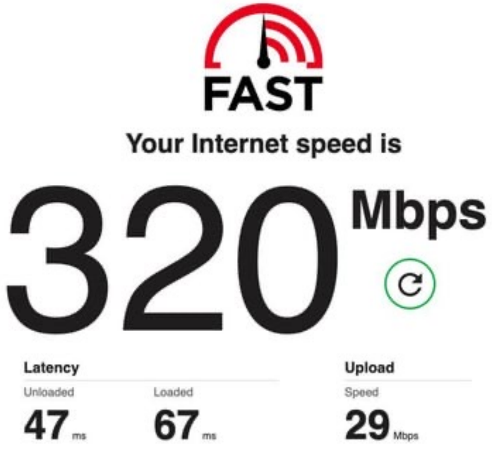
①Constellation Consistency	②System and Networking Stack Realism	③Flexible and Scalable Environment
Spatial and temporal characteristics of a real constellation	Run user-defined system codes and network functionalities like in a real system	Flexibly support various network topologies and diverse test requirements

Problems with Existing ENE Approaches

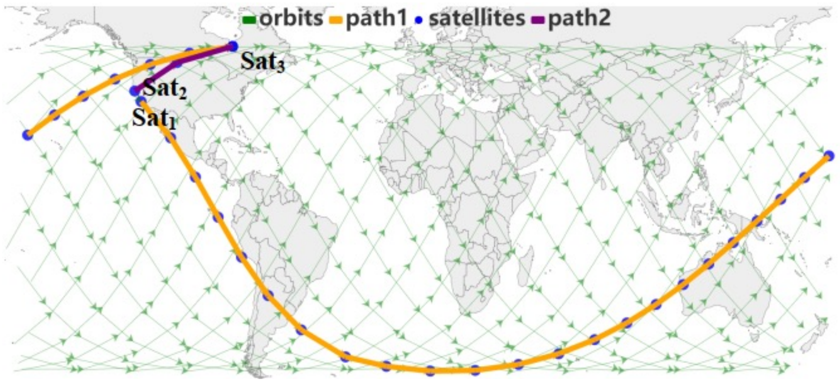
Tsinghua University

①Constellation Consistency	②System and Networking Stack Realism	③Flexible and Scalable Environment
Spatial and temporal characteristics of a real constellation ✓	Run user-defined system codes and network functionalities like in a real system ✓	Flexibly support various network topologies and diverse test requirements ✗

- **Approach I: conducting experiments in a live satellite network**
  - Flexibility and scalability are limited
  - End-host test only, and it is difficult to conduct various *what-if* experiments



iPerf benchmark? Sure!



Benchmarking my new routing protocol upon 4400 LEO satellites? Emm …

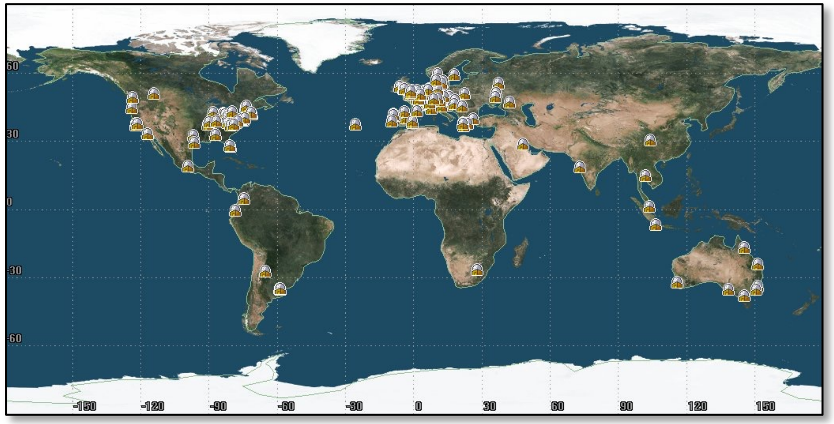
Problems with Existing ENE Approaches

Tsinghua University

①Constellation Consistency	②System and Networking Stack Realism	③Flexible and Scalable Environment
Spatial and temporal characteristics of a real constellation ✔	Run user-defined system codes and network functionalities like in a real system ✘	Flexibly support various network topologies and diverse test requirements ✔

■ Approach II: network simulators

- Realism is limited, since it runs abstractions instead of real applications



STK

GMAT

ns-3

Hypatia [IMC' 20]
StarPerf [ICNP' 20]

Problems with Existing ENE Approaches

清華大學 Tsinghua University

①Constellation Consistency	②System and Networking Stack Realism	③Flexible and Scalable Environment
Spatial and temporal characteristics of a real constellation ✗	Run user-defined system codes and network functionalities like in a real system ✓	Flexibly support various network topologies and diverse test requirements ✓

- **Approach III: network emulators**
  - VM- or container-based emulation
  - Existing emulators can not mimic dynamic behaviors of LEO constellations
  - Some of them are also difficult to scale to very large constellation emulation (e.g. thousands of LEO satellites)

Mininet
> sudo mn

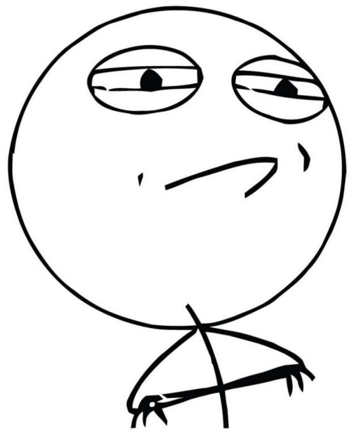
DieCast[TOCS' 11]: VM-based emulation
Etalon[NSDI' 20]: container-based emulation

Our Goal

Tsinghua University

①Constellation Consistency	②System and Networking Stack Realism	③Flexible and Scalable Environment
Spatial and temporal characteristics of a real constellation ✓	Run user-defined system codes and network functionalities like in a real system ✓	Flexibly support various network topologies and diverse test requirements ✓

Can we build an ENE simultaneously satisfying all the above requirements?



CONTENTS

Our Approach

Tsinghua University

- **StarryNet:** a new evaluation framework for ISTN experiments
- **Key idea:** building a **data-driven**, **hybrid** ENE

Public information from real satellite Internet constellations

Regulator

Federal Communications Commission

Consultative Committee for Space Data Systems (CCSDS)

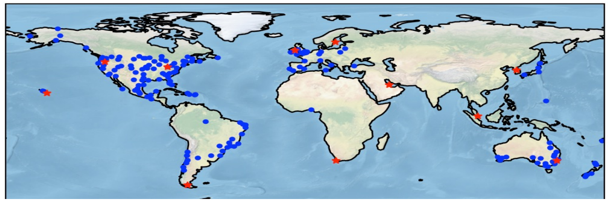
Satellite operator

STARLINK

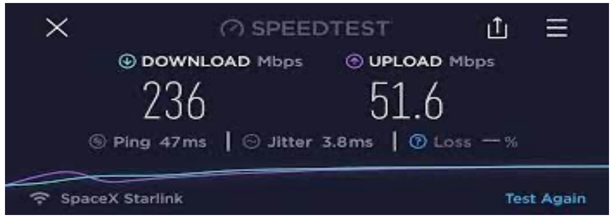
project kuiper

OneWeb

Ground station operator



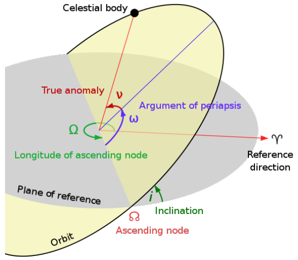
User statistics

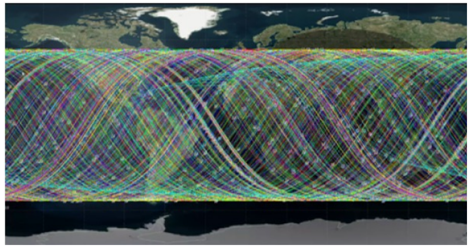


drive

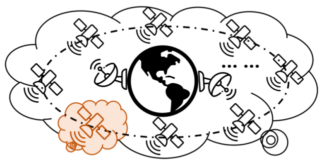
Combining model-based simulation, emulation and satellite hardware

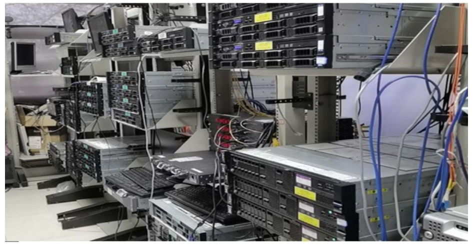
Constellation-model-based simulation





Large-scale emulation cluster





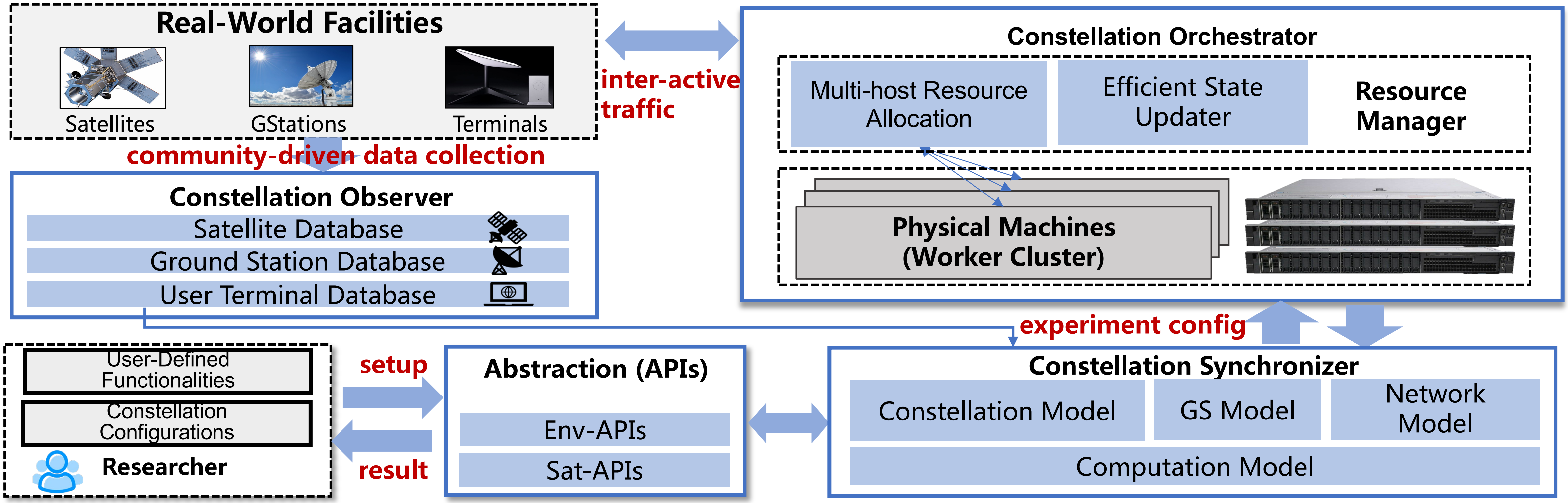
Satellite hardware (e.g. low-power processor)

StarryNet Architecture

System overview

Constellation Observer | **Constellation Synchronizer**

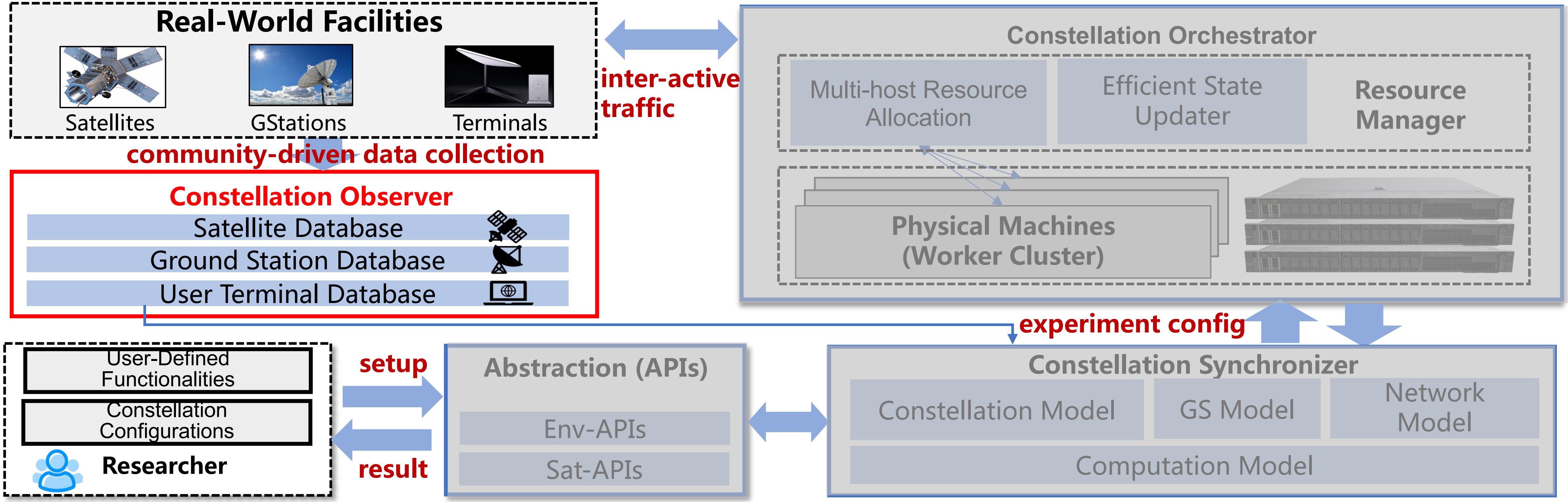
Constellation Orchestrator | **Abstraction (APIs)**



StarryNet Design Details

■Constellation Observer

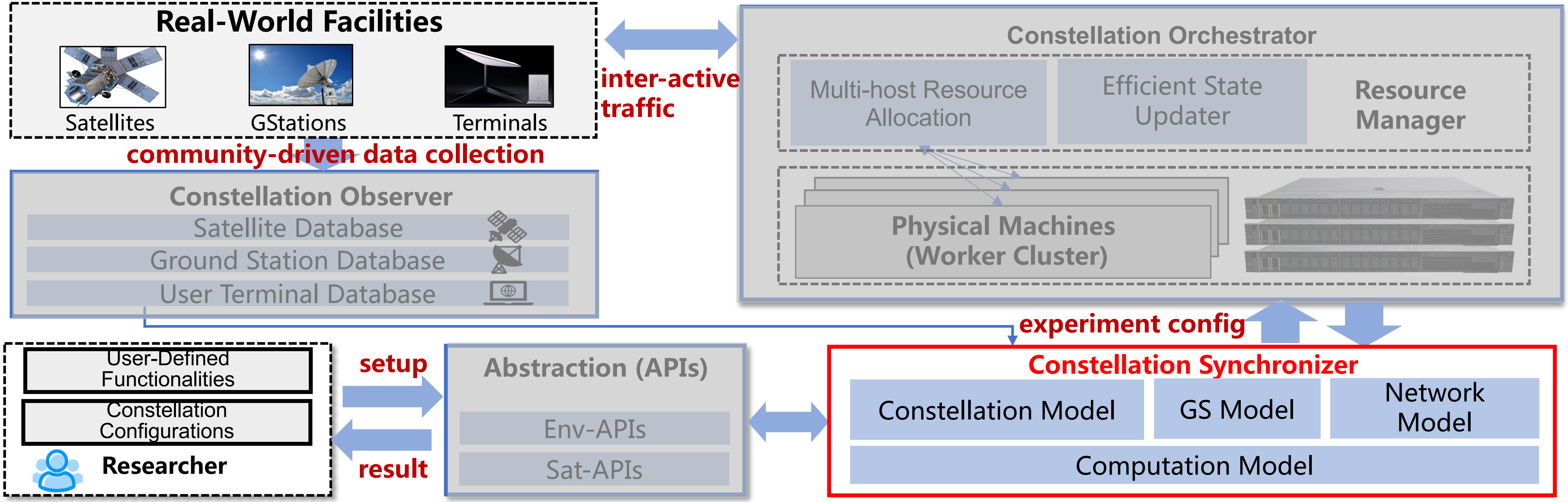
- **Crowd-sourcing approach** to collect public information
- Databases to store constellation-relevant data (e.g. constellation elements)
- Exploiting **multidimensional, realistic data** to support ENE creation



StarryNet Design Details

Constellation Synchronizer

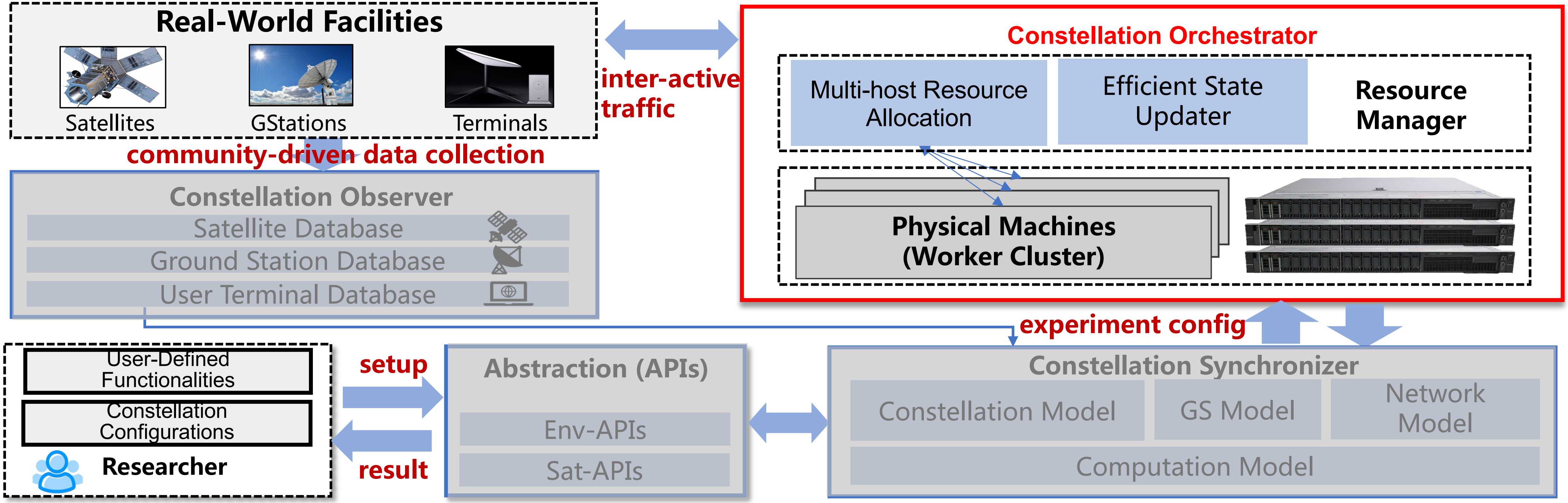
- **Building a series of models** to characterize ISTN network features
- Driven by **realistic constellation information** and **user-defined experiment requirements** to calculate **spatial and temporal behaviors**



StarryNet Design Details

Constellation Orchestrator

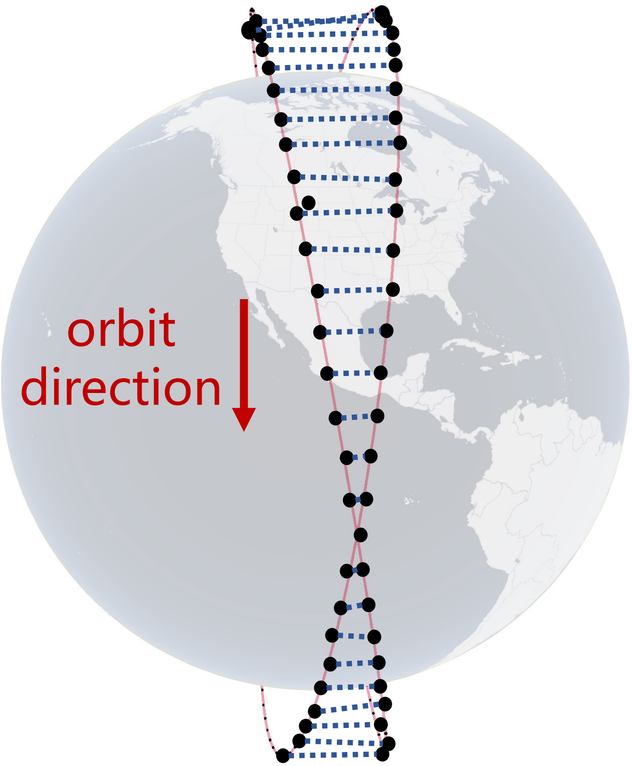
- **Container-based** emulation on physical machines
- Each container mimic a satellite/ground-station/terminal
- Support **flexible computation and network capability** in each node

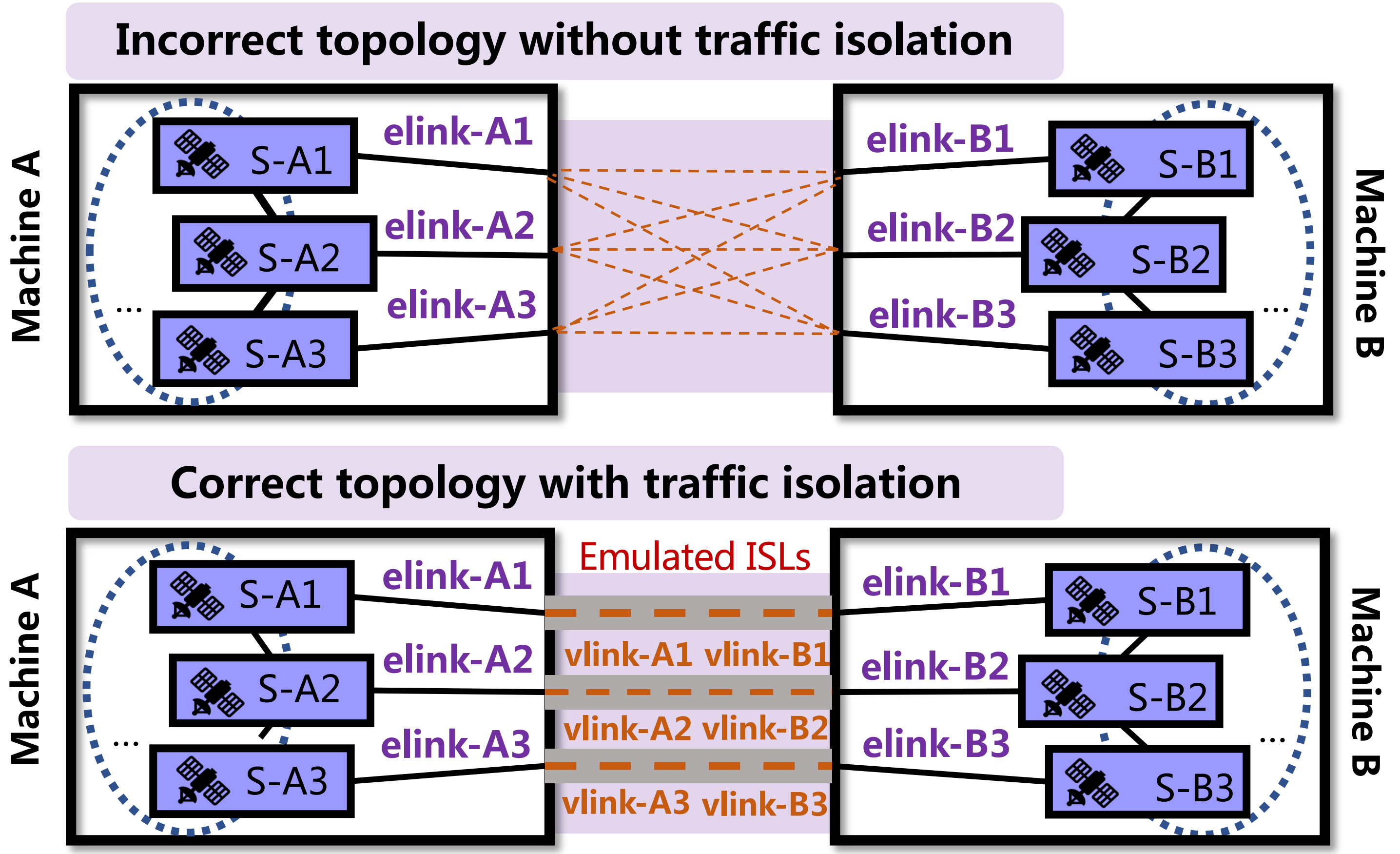


StarryNet Design Details

Constellation Orchestrator

- **Multi-machine extension** for large-scale mega-constellation
- Leverage VLAN-based traffic isolation to build correct network topology





Framework Usage: An Example

Tsinghua University

① Self-defined program

```
# geo_routing.py
from lib_starrynet import *;
def geocast_next_hop(dst_addr):
    # Obtain adjacent satellites info
    n_sats = sn_get_sat_neighbors()
    # Find the sat closest to dst
    for sat in n_sa
        if dis(sat,
            < dis(ne
            next_sat
    return next_sat
```

② Configuration file

```
"starlink": [ #starlink.json
  {  "name": "SL-Phase-I-shell-I",
     "altitude": "550km",
     "inclination": "53.0",
     "plane_count": "72",
     "satellites_per": "22" }]
                    : [ #gs.json
                    go",
                    1.850",
                    -87.650",
                    .144683km"},...]
```

③ Shell commands

```
# listen on manager machine
@manager:/$ sn manager init --m-addr=192.168.0.1
# on each worker machine, join the framework
@worker: /$ sn worker join --m-addr=192.168.0.1
# on manager machine, load manifest files and create the ENE
@manager:/$ sn create --name sl_cons -c 'starlink.json' -gs 'gs.json'
# start the ENE for 3600 seconds
@manager:/$ sn start sl_cons --duration=3600
# run user-specific program in all satellites in the first orbit
@manager:/$ sn cmd sl_cons.orbit[0] python geo_routing.py
```

CONTENTS

Evaluation Setup

■ **StarryNet implementation**
- Eight high-performance DELL R740 servers in a cluster. Each one with 2*Intel Xeon 5222 (4 cores @ 3.8GHz), and 8*32GB DDR RAM
- Based on Docker Container, OpenvSwitch, tc, *etc.*

■ **Open data**
- CeleTrak[1] (orbital information), FCC filing … *etc.*

■ **Evaluation and Use Case**
- Ability to satisfy various experimental requirements for ISTNs
- Fidelity analysis
- Case studies

[1] https://www.celestrak.com/

Various Constellation Configurations

Tsinghua University

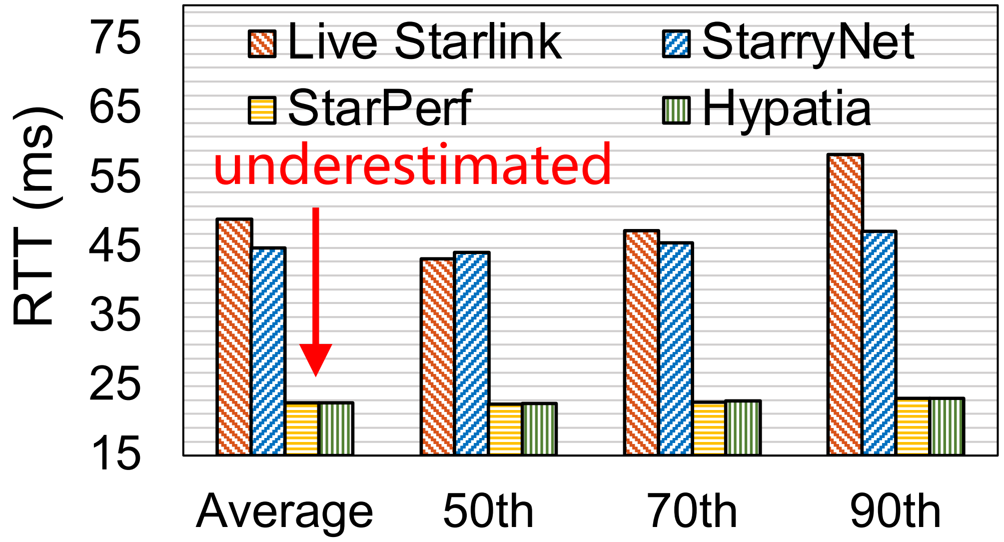
STARRYNET is flexible to scale to various constellation configurations with different network topologies

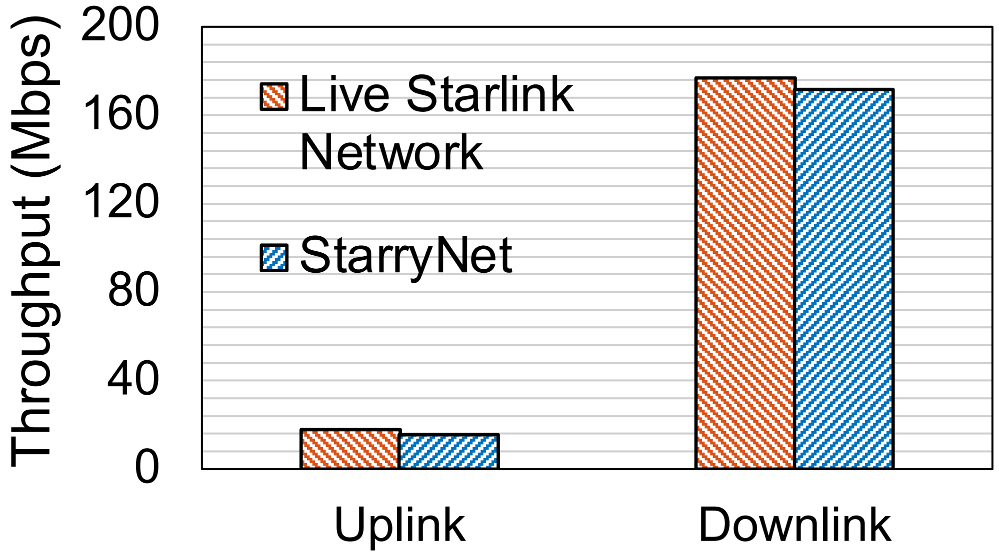
Metrics / Constellation	Height (km)	Constellation Size (number of satellites)	Creation Time (min) Nodes/Links/Total			Avg. CPU (%) Interval = 1/2/3 (s)			Avg. Memory (%) Interval = 1/2/3 (s)			Minimum # of Required Workers
Starlink S1 (72*22, 53°)	550	1584	5.9	4.6	10.5	7.2%	7.0%	6.3%	3.9%	3.5%	3.4%	2
Starlink S2 (72*22, 53.2°)	540	1584	5.9	4.6	10.5	7.2%	7.0%	6.3%	3.9%	3.5%	3.4%	2
Starlink S3 (36*20, 70°)	570	720	3.0	2.1	4.9	1.2%	1.1%	1.0%	2.7%	2.6%	2.6%	1
Starlink S4 (6*58, 97.6°)	560	348	1.9	1.3	3.2	1.0%	1.0%	1.0%	2.7%	2.6%	2.4%	1
Starlink S5 (4*43, 97.6°)	560	172	1.6	1.2	3.2	1.0%	1.0%	1.0%	2.3%	2.3%	2.3%	1
Starlink Full (4408 satellites)	hybrid	4408	13.3	7.9	21.2	39.6%	37.0%	34.3%	10.4%	9.1%	8.9%	7
Kuiper K1 (34*34, 51.9°)	630	1156	4.4	3.8	8.2	2.6%	2.4%	2.3%	3.8%	3.5%	3.2%	2
Kuiper K2 (36*36, 42°)	610	1296	4.7	4.2	8.9	3.9%	3.6%	3.2%	4.0%	3.6%	3.5%	2
Kuiper K3 (28*28, 33°)	590	784	3.2	2.4	5.6	1.3%	1.2%	1.2%	2.7%	2.6%	2.6%	2
Kuiper Full (3236 satellites)	hybrid	3236	5.7	4.8	10.5	24.6%	23.9%	23.2%	6.3%	6.2%	6.2%	6
Telesat T1 (27*13, 98.98°)	1015	351	1.9	1.3	3.2	1.0%	1.0%	1.0%	2.6%	2.5%	2.4%	1
Telesat T2 (40*33, 50.88°)	1325	1320	4.8	4.2	9.0	3.9%	3.7%	3.3%	4.0%	3.6%	3.5%	2
Telesat Full (1671 satellites)	hybrid	1671	3.1	2.4	5.5	7.2%	7.0%	6.4%	4.2%	3.7%	3.6%	3

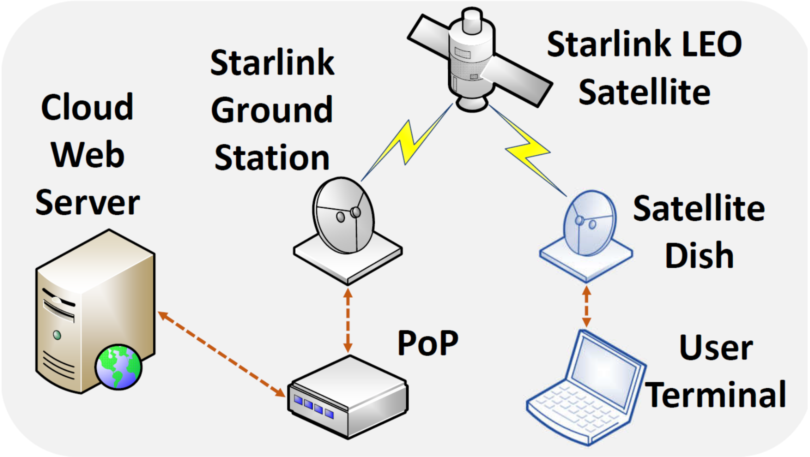
Fidelity Analysis

Tsinghua University

Network performance under the same bent-pipe topology compared with live Starlink and other simulation tools



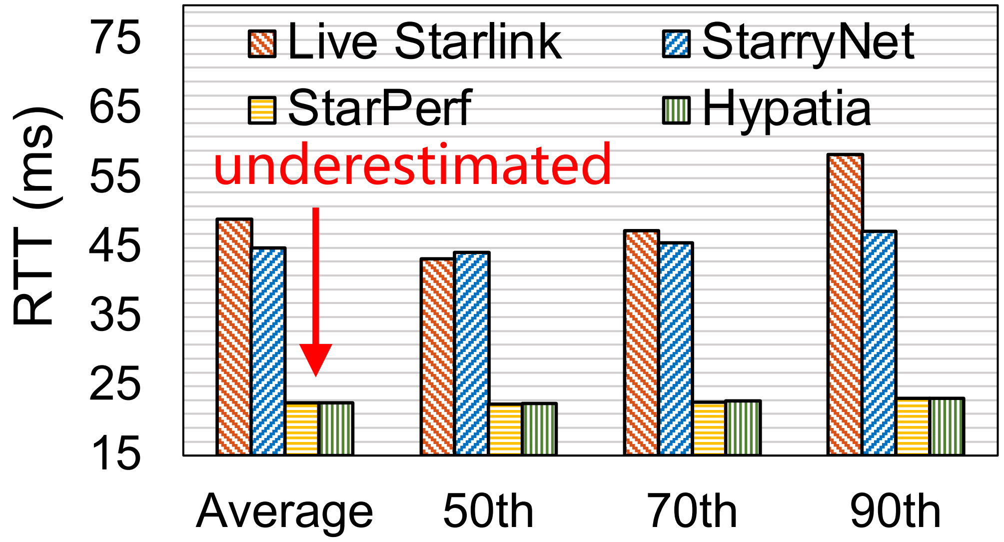


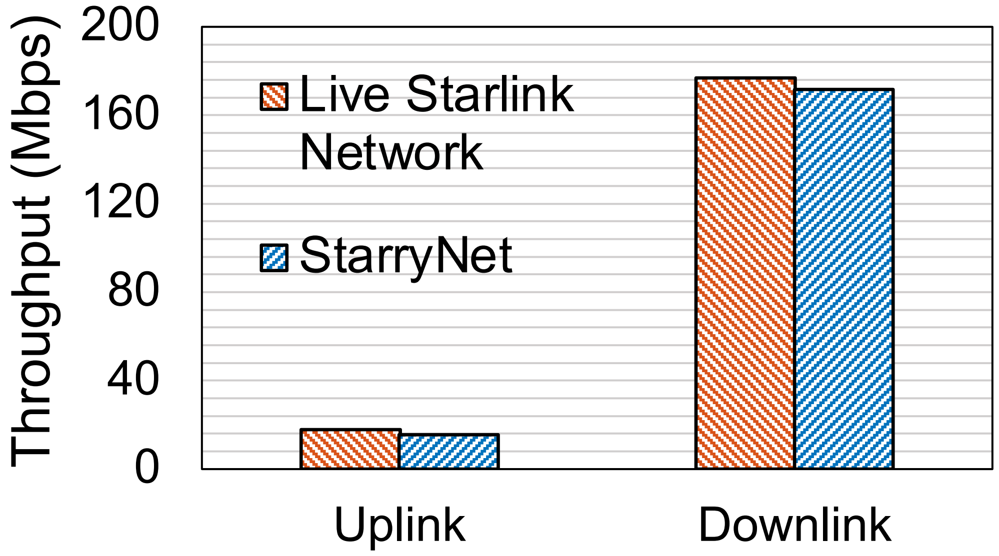


Fidelity Analysis

Tsinghua University

Network performance under the same bent-pipe topology compared with live Starlink and other simulation tools



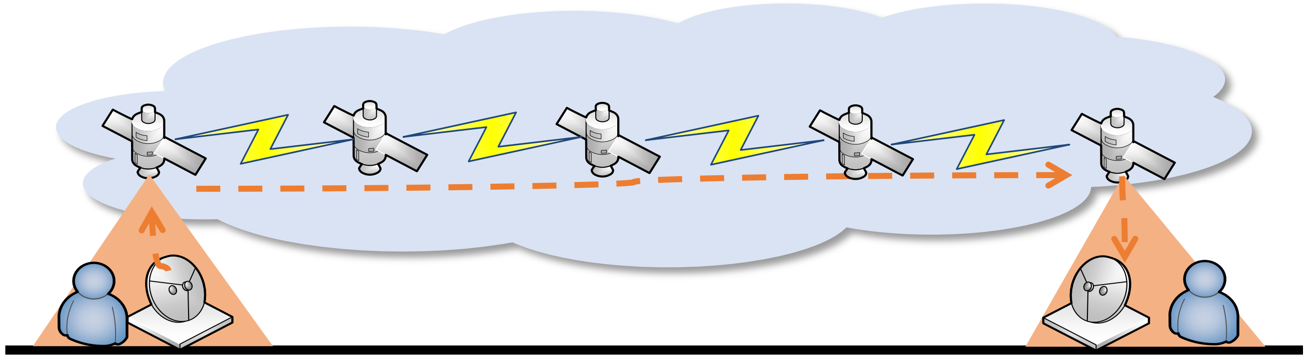


StarryNet achieves acceptable fidelity

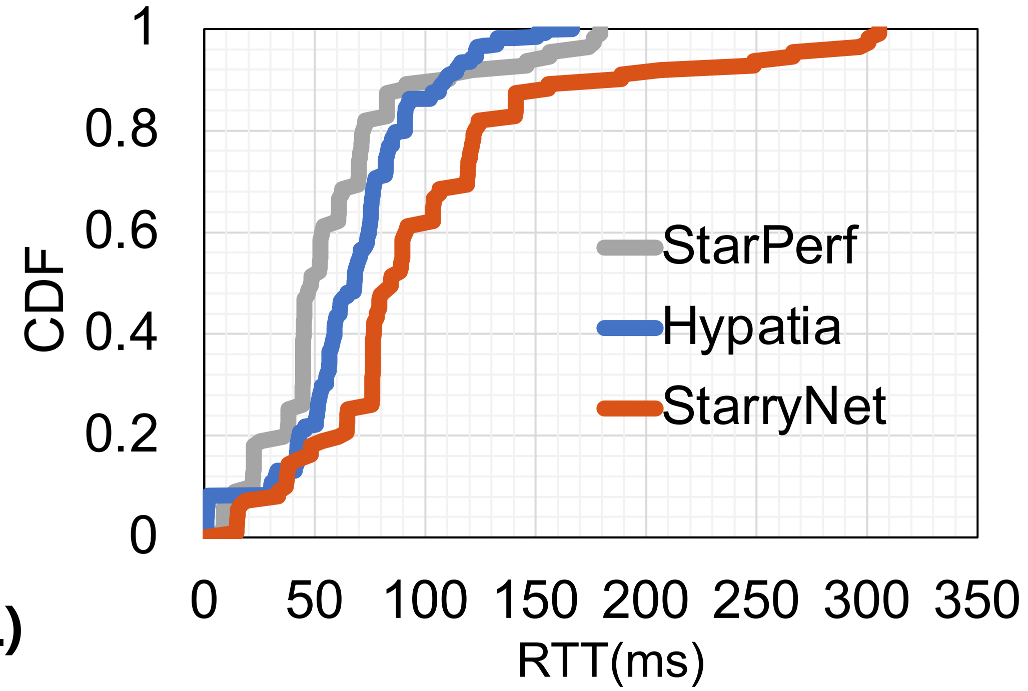
- **Similar latency performance** to live Starlink measurements
- Accurately emulating the **bandwidth** of a live ISTN

Fidelity Analysis

Network performance with ISLs
compared with other two simulation tools



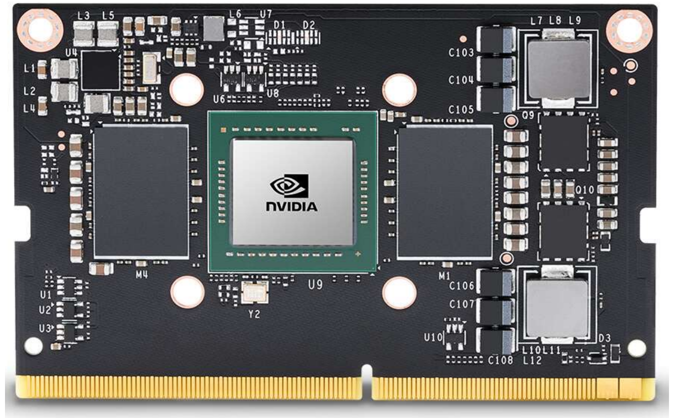
End to end RTT through inter-satellite links (ISL)



- At this time it is difficult to measure real ISL performance
- We analyze the results as compared with other simulators
- Similar results but involve additional **system-level overhead**

Fidelity Analysis

Emerging satellites are equipped with evolved computation capabilities to support various on-board applications



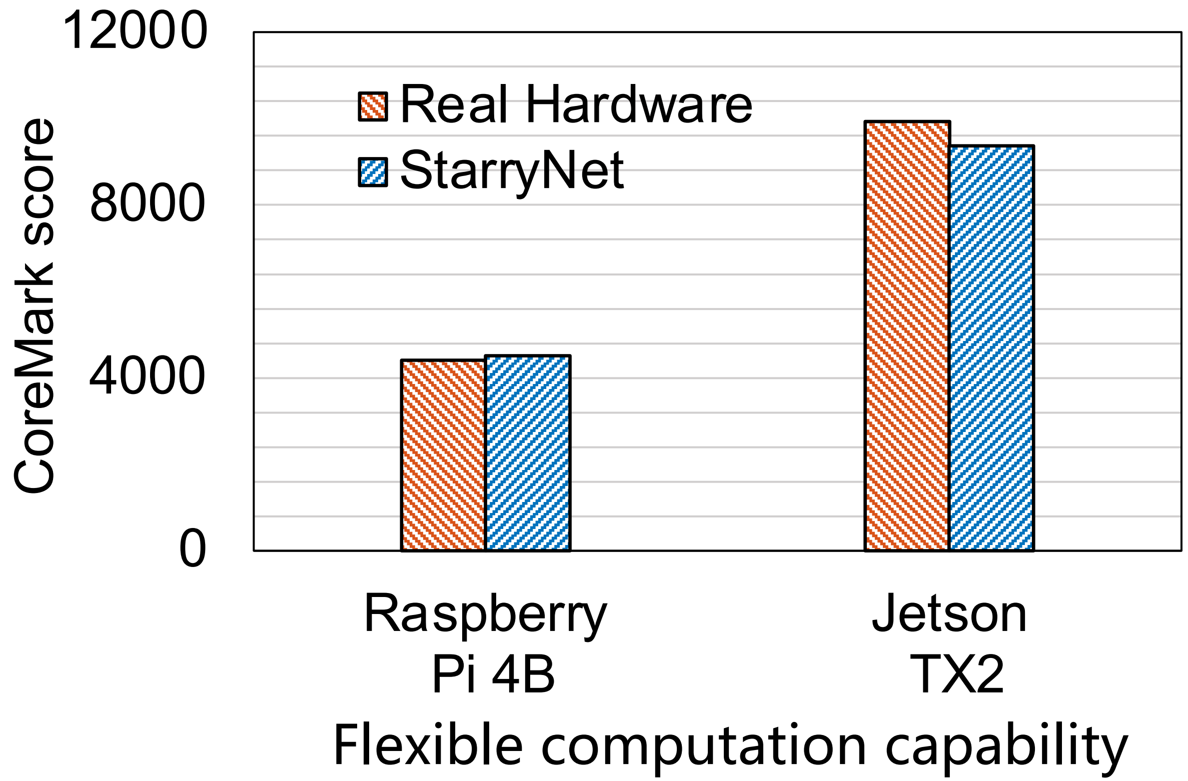
Orbital edge computing (OEC) uses Jetson TX2 to enable on-board AI capability

European Space Agency (ESA) uses low-power Raspberry Pi for on-board missions

StarryNet can be configured to mimic various computation capabilities on-demand

Orbital Edge Computing: Nanosatellite Constellations as a New Class of Computer System, ASPLOS 2020.
https://www.esa.int/Education/AstroPI

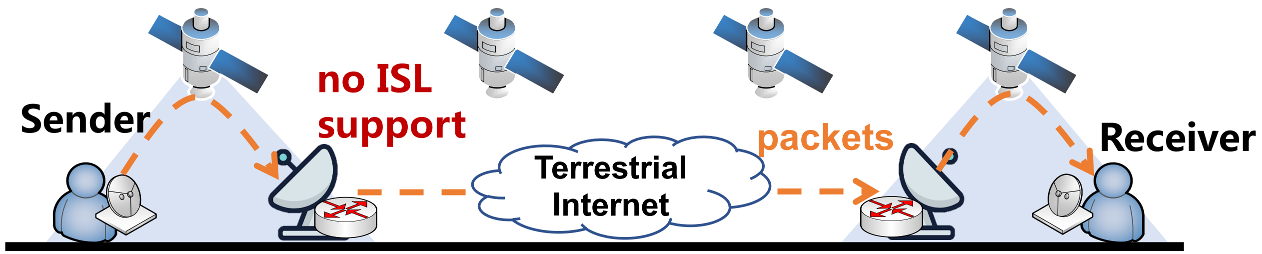
Fidelity Analysis



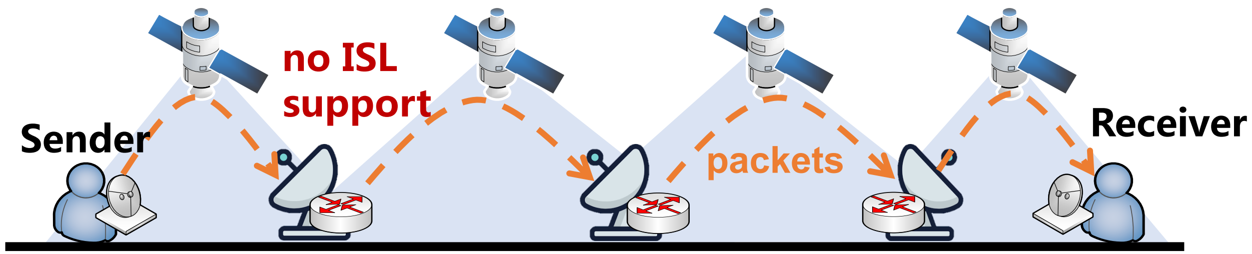
StarryNet can be configured to mimic various computation capabilities on-demand

Case Study I: Interconnecting LEO Satellites and Terrestrial Facilities

Tsinghua University

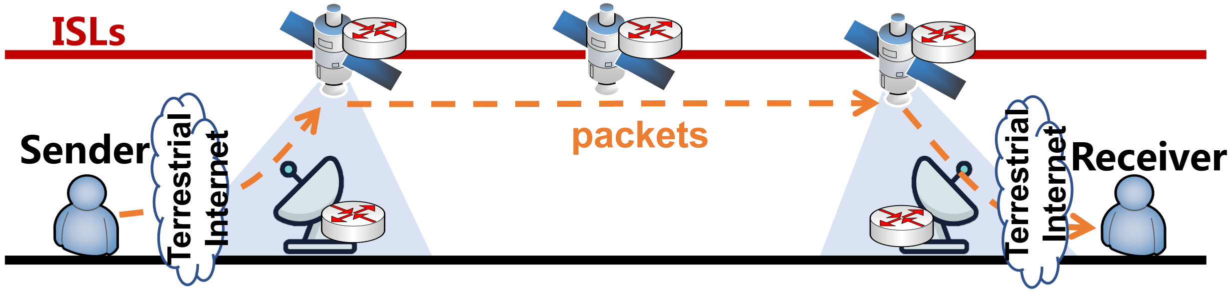


SRLA: satellite relays for last-mile accessibility

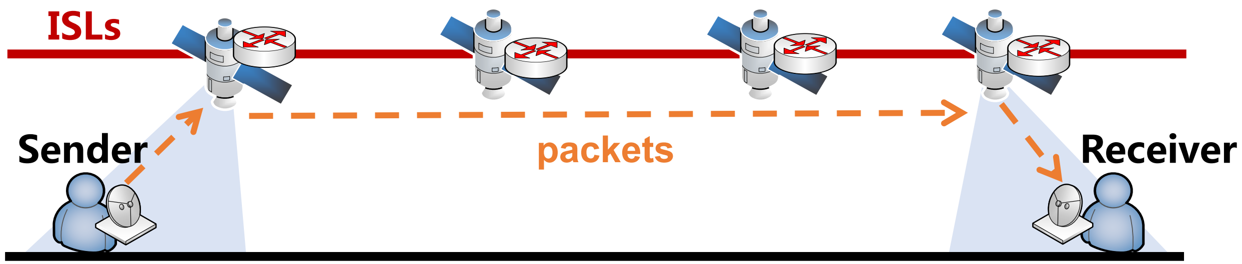


SRGS: satellite relays for ground station networks

Exploring the design-space for various space-ground integration methodologies



GSSN: ground station access for satellite networks



DASN: satellite networks directly accessed by terrestrial users

Case Study I: Interconnecting LEO Satellites and Terrestrial Facilities

清華大學 Tsinghua University

StarryNet supports realistic routing and data transmission for mega-constellations

Latency comparison | Network reachability comparison | Addressing and cost comparison

Design	Average end-to-end latency and its breakdown (ms)				Reachability	Frequent Address Update	Operating Cost		
	Inter-Satellite	Space-Ground	Ground	Toal			GS	Terminal	ISLs
SRLA	0	76.25	107	183.25	97.00%	✗	✓	✓	✗
SRGS	0	313.39	0	313.39	51.00%	✗	✓	✓	✗
GSSN	48.46	38.45	20	106.91	57.40%	✗	✓	✗	✓
DASN	48.46	37.65	0	86.11	97.50%	✓	✗	✓	✓

Conclusions

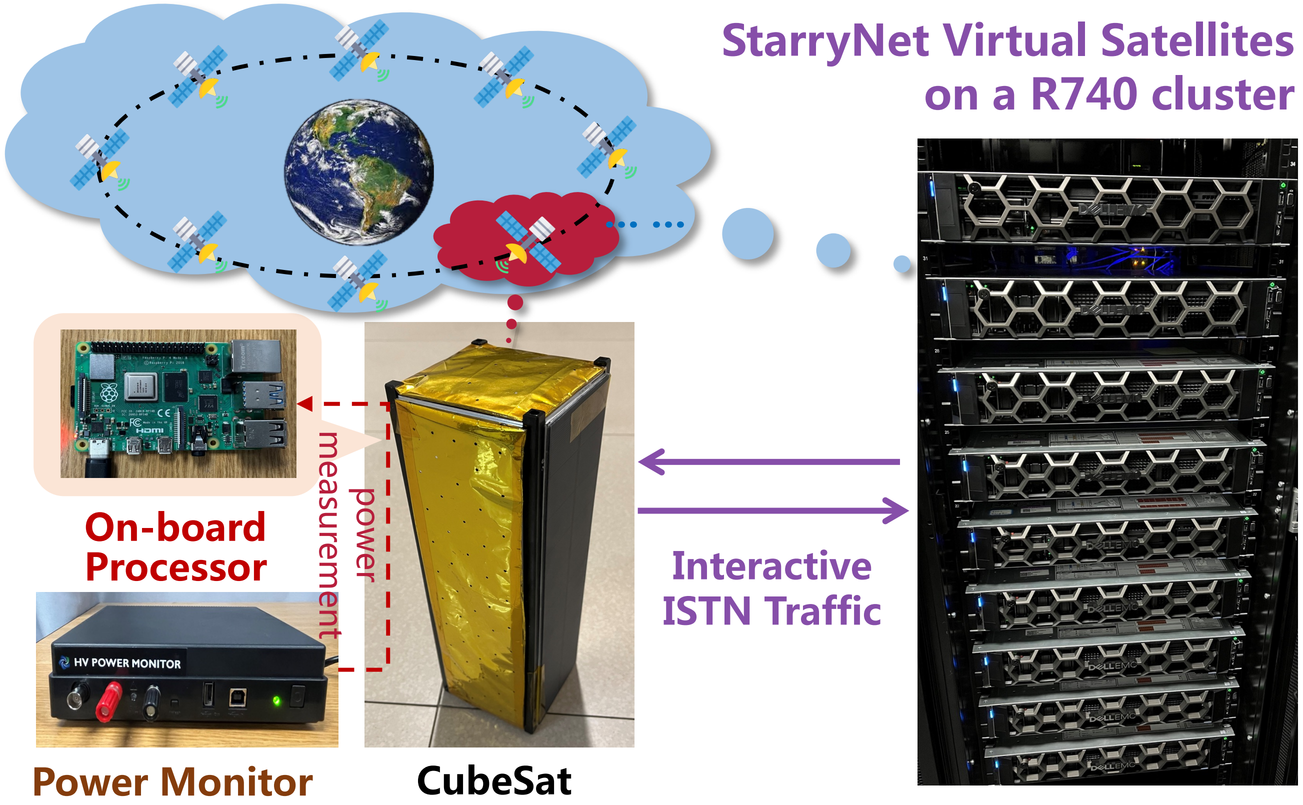
- An obvious latency reduction accomplished by ISLs
- Reachability discrepancy caused by handovers and uneven GS distributions
- Deployment and costs vary a lot

Case Study II: Hardware-in-the-loop Testing

Tsinghua University

STARRYNET supports a hybrid deployment and evaluates real system effects for user-defined functionalities

A number of virtual, emulated nodes + 1 real prototype



- **Evaluate system-level effects of a new ISTN network protocol or functionality**
  - Link advertisement overhead of a new routing protocol
  - Power consumption
  - CPU usage
  - Memory overhead … …

State	Idle	Routing convergence	Transmission rate (Mbps)			
			100	250	500	750
Power (W)	2.83	3.22	4.6	5.0	5.4	5.5

CONTENTS

Conclusion

Tsinghua University

- **Existing tools fail to guarantee realism, flexibility, and low-cost simultaneously**
- **StarryNet is able to achieve the goal by**
  - Integrating real constellation-relevant information, orbit analysis, etc.
  - Container-based large-scale emulations
  - Low-cost usage and open APIs
- **Evaluation results show that StarryNet**
  - Achieves high-fidelity to real measurements
  - Supports various ISTN experiments flexibly

Thank you!

Q&A

Zeqi Lai, Hewu Li, Yangtao Deng, Qian Wu, Jun Liu, Yuanjie Li,
Jihao Li, Lixin Liu, Weisen Liu, Jianping Wu

Contact

- https://github.com/SpaceNetLab/StarryNet
- zeqilai@tsinghua.edu.cn
- yangtaodeng@gmail.com

Read our paper!