ALTO/FTS: FTS Control with Deeper Network Visibility

Presenters: Mario Lassnig, Mihai Patrascoiu,
Jordi Ros-Giralt, Y. Richard Yang

On behalf of team

03/27-31/2023
XRootD and FTS Workshop @ Jozef Stefan Institute

High-Level Motivation and Goal

User Demands Evolve

- Higher **efficiency** to use all available resources to satisfy increasing traffic volumes
- Higher **flexibility** to share the same infrastructure among **multiple** user experiments

Network Infrastructure Capabilities Evolve

- Increasingly more advanced control network behaviors (e.g., SENSE/AutoGOLE, NOTED)
- Protocol providing **visibility** of network state (e.g., IETF ALTO Protocol RFC7285)

System Architectures Evolve

- Traditional networking
- Software defined networking (SDN)
- Application-defined infrastructure/network

Project goal: Extend FTS as an **application-defined networking controller**, through deeper network (resource) state **visibility**, to improve its **efficiency** and **flexibility**.

Background: Application-Layer Traffic Optimization (ALTO)

- ALTO is an effort of the ALTO working group in the Transport Area of Internet Engineering Task Force (IETF)
- ALTO high-level goal: provide a standard for applications and networks to work together to optimize both network and application performance
- Two core components:
  - Abstractions of network state/services
  - Transport and discovery of abstractions

Problem Statement (RFC 5693)	Xdom Service Discovery (RFC 8686)	Base Protocol (RFC 7285)	Multicost (RFC 8189)	Path Vector (RFC 9275)
Requirements (RFC 6708)	Server Discovery (RFC 7286)	SSE/Incr Update (RFC 8895)	Cost Calendar (RFC 8896)	Cost Metrics (RFC editor)
Comcast Deployment (RFC 5632)	Deployment (RFC 7971)		CDNi (RFC 9241)	Unified Properties (RFC 9240)

ALTO Abstraction Example: Path Vector

```
POST /endpointcost/pv HTTP/1.1
Host: alto.example.com
Accept: multipart/related;
        type=application/alto-endpointcost+json,
        application/alto-error+json
Content-Length: 362
Content-Type: application/alto-endpointcostparams

{
  "cost-type": {
    "cost-mode": "array",
    "cost-metric": "ane-path"
  },
  "endpoints": {
    "srcs": [
      "ipv4:192.0.2.34",
      "ipv6:2001:db8::3:1"
    ],
    "dsts": [
      "ipv4:192.0.2.2",
      "ipv4:192.0.2.50",
      "ipv6:2001:db8::4:1"
    ]
  },
  "ane-property-names": [
    "max-reservable-bandwidth",
    "persistent-entity-id"
  ]
}
```

```
HTTP/1.1 200 OK
Content-Length: 1433
Content-Type: multipart/related; boundary=example-2;
             type=application/alto-endpointcost+json

--example-2
Content-ID: <ecs@alto.example.com>
Content-Type: application/alto-endpointcost+json

{
  "meta": {
    "vtags": {
      "resource-id": "endpoint-cost-pv.ecs",
      "tag": "bb6bb72eafe8f9bdc4f335c7ed3b10822a391cef"
    },
    "cost-type": {
      "cost-mode": "array",
      "cost-metric": "ane-path"

    }
  },
  "endpoint-cost-map": {
    "ipv4:192.0.2.34": {
      "ipv4:192.0.2.2":   [ "NET3", "L1", "NET1" ],
      "ipv4:192.0.2.50":   [ "NET3", "L2", "NET2" ]
    },
    "ipv6:2001:db8::3:1": {
      "ipv6:2001:db8::4:1": [ "NET3", "L2", "NET2" ]
    }
  }
}
```

```
--example-2
Content-ID: <propmap@alto.example.com>
Content-Type: application/alto-propmap+json

{
  "meta": {
    "dependent-vtags": [
      {
        "resource-id": "endpoint-cost-pv.ecs",
        "tag": "bb6bb72eafe8f9bdc4f335c7ed3b10822a391cef"
      },
      {
        "resource-id": "ane-props",
        "tag": "bf3c8c1819d2421c9a95a9d02af557a3"
      }
    ]
  },
  "property-map": {
    ".ane:NET1": {
      "max-reservable-bandwidth": 50000000000,
      "persistent-entity-id": "ane-props.ane:MEC1"
    },
    ".ane:NET2": {
      "max-reservable-bandwidth": 50000000000,
      "persistent-entity-id": "ane-props.ane:MEC2"
    },
    ".ane:NET3": {
      "max-reservable-bandwidth": 50000000000
    },
    ".ane:L1": {
      "max-reservable-bandwidth": 10000000000
     },
    ".ane:L2": {
      "max-reservable-bandwidth": 15000000000
     }
  }
```

- More details see https://datatracker.ietf.org/doc/html/draft-ietf-alto-path-vector-21#section-8.1

FTS Functions/Desired Properties/Mechanisms

Functions: Transport Scheduling, decides

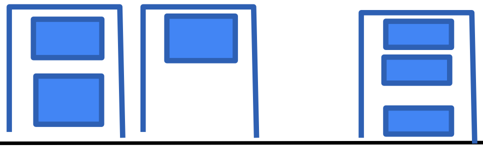
- when to start the transport of a transfer request
- with how much transport resource

Desired Properties:

- Efficiency: effectively utilize transport resources
- Flexibility/fairness: share resources with control

Basic (Optimizer Control) Mechanisms:

- Keeps transfer queue for each src/dst pair (call **pipe**)
- Adjusts # concurrent TCP connections per pipe
- Dispatches transfer if allowed by concurrency level



Data-Intensive Workflows

Data Management / Transfer Orchestration (e.g., Rucio)

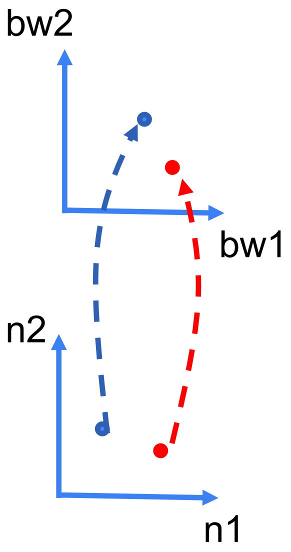
Transfer Scheduling (e.g., FTS)

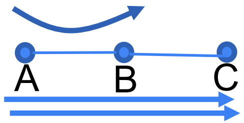
Transfer Data Plane (e.g., GridFTP, XRootD, HTTP)

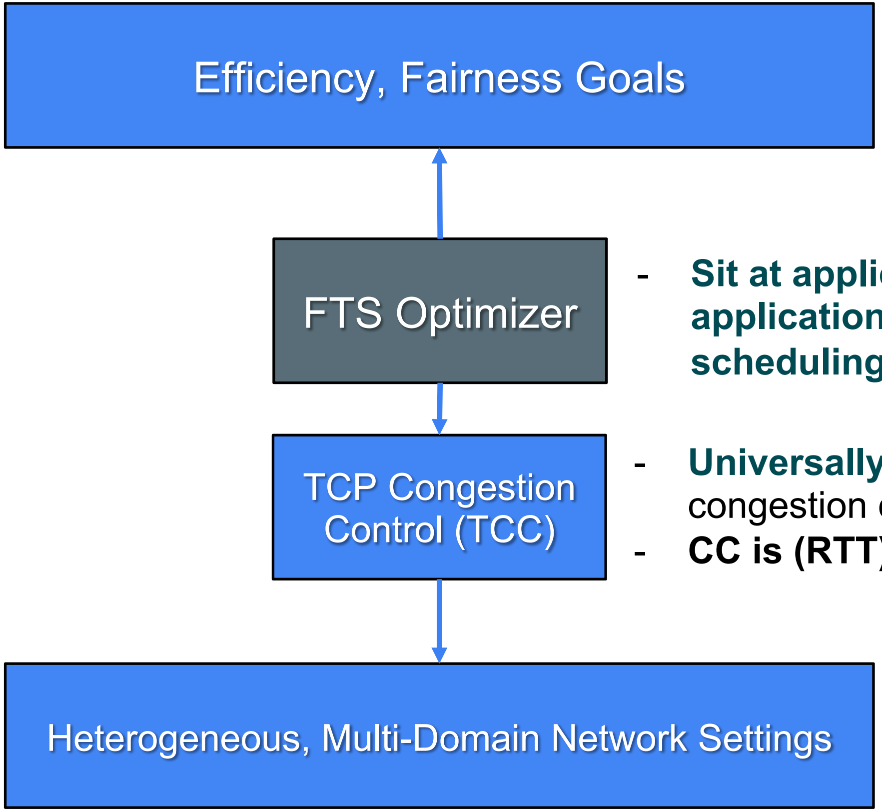
Internet Transport Layer (e.g., TCP, TCP/Cubic, BBR)

Networking Layer (e.g., traditional networking, AutoGOLE/SENSE, NOTED, Programmable net)

FTS as an Elegant Base Architecture for Application-Defined Networking/Infrastructure







- **Sit at application level, can understand application semantics and conduct scheduling**

- **Universally available**, fully **distributed** congestion control (CC)
- **CC is (RTT) fast, efficient, robust**

Gap of Default FTS Control

Protocol 1 FTS Model Analyzed (Called for High Success Rate)

```
1:  Define RL(x) = round(log_B(x))
2:  procedure OPTIMIZEGOODSUCCESSRATE(state)
3:      if cur.ema < prev.ema then
4:          if RL(cur.ema) < RL(prev.ema) then
5:              decision = prevValue - decreaseStepSize
6:          else
7:              decision = prevValue
8:          end if
9:      else if cur.ema > prev.ema then
10:         decision = prevValue + increaseStepSize
11:     else                                        ▷ emas are equal
12:         decision = prevValue + increaseStepSize
13:     end if
14: end procedure
```

Simple conn# limit, Semi Zero-Order Gradient Alg
Optimizing for Each Pipe Alone

Keep track of the exponential moving average (EMA) of throughput.

$$E_i(t+1) = \alpha T_i(t+1) + (1-\alpha) E_i(t)$$

Update the number of connections based on EMA.

$$n_i(t+1) = \begin{cases} n_i(t) - 1 & RL_B(E_i(t+1)) < RL_B(E_i(t));\ \text{Line 4} \\ n_i(t) + 1 & E_i(t+1) \geq E_i(t);\ \text{Lines 9,11} \\ n_i(t) & \text{else} \end{cases}$$

controllability gap

Theorem 4.2 (Conservation Theorem). *Let $K = \max \frac{M_i}{m_i}$. Then as long as $B > (1 - \alpha + \alpha K^2)$, the quantity*

$$V_t(t) = n_i(t) - \mathrm{round}(\log_B(E_i(t)))$$

only ever stays constant or increases.

Implicit upper limit control,
not explicit resource structure model

efficiency gap

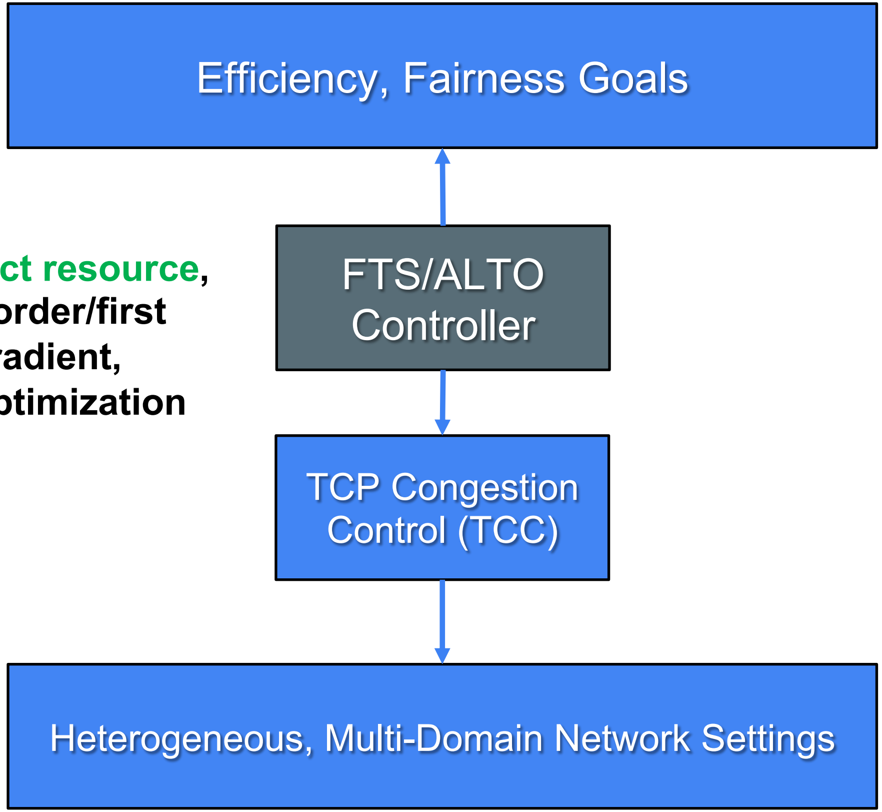
Theorem: In a **Throughput-Deterioration Model, semi zero order** will achieve throughput that is **<=** 1/√B of the **optimal** (under default settings).

Can be suboptimal in some cases

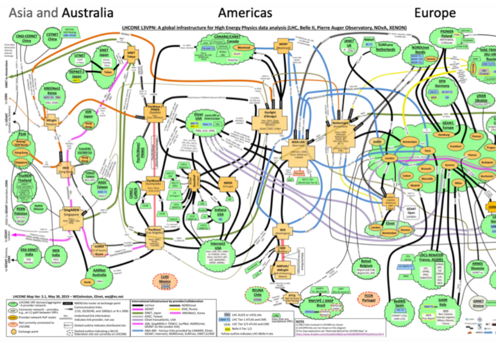
FTS => FTS/ALTO

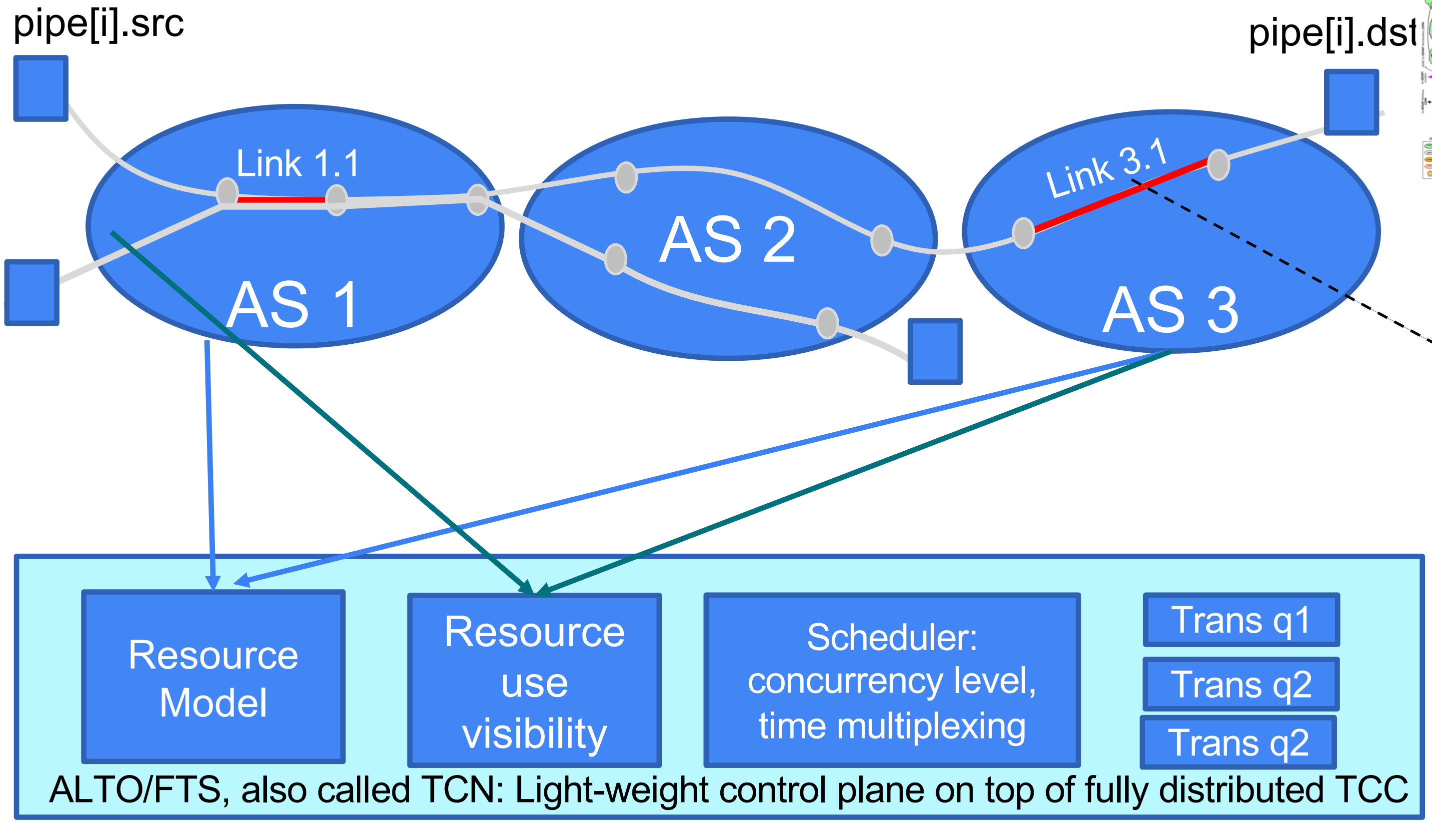
Simple conn# limit, semi zero-order gradient, optimizing for each pipe alone

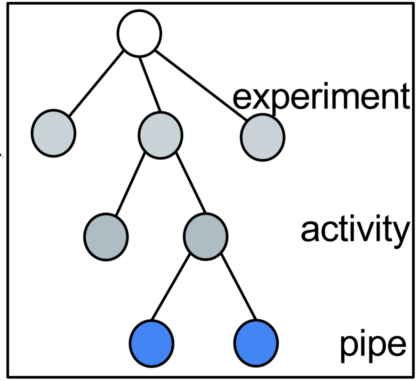
Flexible, direct resource, fully zero-order/first order gradient, with joint optimization

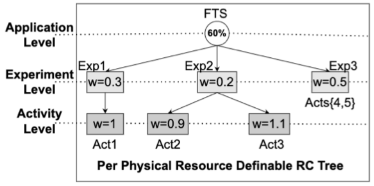


ALTO/FTS Architecture









Scheduler as an Optimization Framework

Driven by an optimization framework, in the form:

max f_{obj}
s.t. direct, flexible resource control as constraints K

Solve the optimization using a **systematic gradient framework.**

Simplified Example for Resource Specification, Control, Visibility

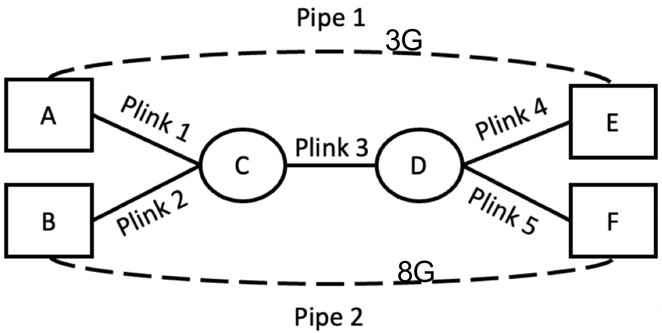
Resource Spec:

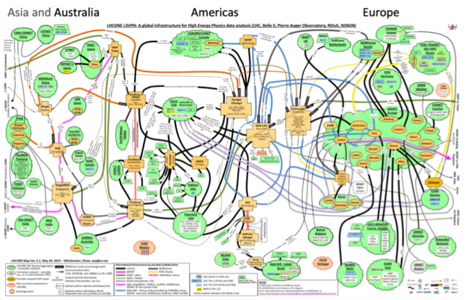
Experiment: Pipe 1, Pipe 2

R1: <Exp. 1, Plink 1> <= 5G

R2: <Exp. 1, Plink 2> <= 10G

R3: <Exp. 1, Plink 3> <= 10G





App Accounting:

Pipe1.traffic = 3G, Pipe2.traffic = 8G

ALTO Provided Mapping:

Pipe 1: {Plink 1, Plink 3, Plink 4}.

Pipe 2: {Plink 2, Plink 3, Plink 5}.

Mapping App on Physical Resources:

Plink1.traffic = 3G, Plink2.traffic = 8G,

Plink3.traffic = Pipe1.traffic + Pipe2.traffic = 11G

Experiment Use vs Resource Spec:

P(R1) = 0 (Plink1 = 3G <= 5G) , P(R2) = 0 (Plink2 = 8G <= 10G)

P(R3) = 2 (Plink3 = 11G > 10G)

ALTO/FTS Control Details (for completeness)

- Integral, quadratic distance functior
- Zero-order stochastic rounding

$$U(\tau) = \left(\sum_{i=1}^{K} w_i \tau_i\right) - \eta \cdot d(\tau, t \cdot K)^2$$

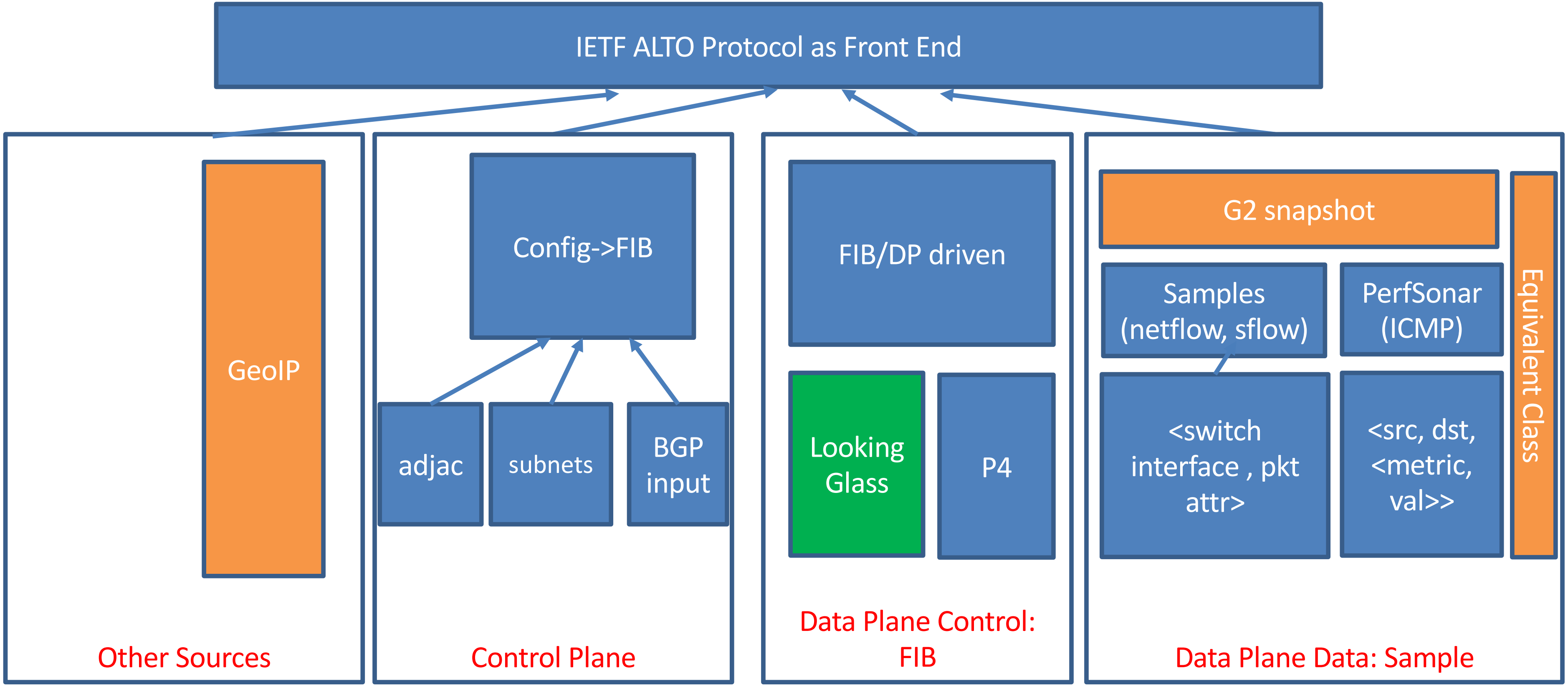
1. Basic Gradient	Gradient of control state n_i: $(\frac{\mathrm{d}a}{\mathrm{d}n_i})$	$\frac{\mathrm{d}a}{\mathrm{d}n_i} = \sum_{j=1}^{K} \frac{\mathrm{d}a}{\mathrm{d}T_j} \cdot \frac{\mathrm{d}T_j}{\mathrm{d}n_i}$
	1.1. $\frac{\mathrm{d}T_j}{\mathrm{d}n_i}$ is the gradient of the bottleneck	If $T_j(n) = \min(f_{j,1}(n), \dots, f_{j,b}(n))$ and $k = \mathrm{argmin} f_{j,k}(n)$, then $\frac{\mathrm{d}T_j}{\mathrm{d}n_i} = \frac{\mathrm{d}f_{j,k}}{n_i}$
	1.2. Decide zero (implicit) or first order (w/ analytical expr)	$\frac{\mathrm{d}f_{j,k}}{\mathrm{d}n_i} = \begin{cases} \text{zero-ord est.} & \text{for blackbox } f_{j,k} \\ \text{first-ord grad.} & \text{otherwise.} \end{cases}$
	1.2a. Zero order estimate	$G(n, z) = \frac{f_{j,k}(n+z) - f_{j,k}(n)}{\lVert z \rVert^2} \cdot z$
	1.2b First order computation	Compute analytical expression: $\frac{\mathrm{d}f_{j,k}}{\mathrm{d}n_i}$
2. Momentum-Based Gradient Acceleration	Compute $g = (\frac{\mathrm{d}a(n)}{\mathrm{d}n_1}, \frac{\mathrm{d}a(n)}{\mathrm{d}n_2}, \dots, \frac{\mathrm{d}a(n)}{\mathrm{d}n_K})$; Update $\mathbf{m} = (1-\alpha)\mathbf{m} + \alpha \cdot (\eta g)$; $n = cur.\mathbf{n} + int(\mathbf{m})$;	
3. Discretize	$int(x) = \begin{cases} \lfloor x \rfloor & \text{with probability } 1 - (x - \lfloor x \rfloor) \\ \lfloor x \rfloor + 1 & \text{with probability } x - \lfloor x \rfloor. \end{cases}$	

ALTO/FTS Benefits (Overview)

- Efficient, shared data transport infrastructure through application-layer scheduling, complementing other layers (e.g., assume given L3)
- Build on FTS

Goal	ALTO/FTS Control Extension	FTS Base
Efficiency	Optimize **global objective** using **full gradient** ⟹ Avoid non-Pareto-optimal solutions.	Optimize the throughput of **each pipe** independently; semi-gradient
	Support both zero-order and **first-order** methods ⟹ Fast convergence.	**Support only zero-order** style gradient estimation.
Control	**Resource model** as constraints ⟹ Support **direct, flexible resource control**	**Resource control focusing on local protection**
App-level control	Allow dependent flows ⟹ Support coflow scheduling.	Focus on individual flows.

Implementation: Visibility



Implementation: First Hop Visibility

Query Example (ECS with path vector extension)

Query/Response

```
→ cat request-cern.json
{
  "cost-type": {
    "cost-metric": "ane-path",
    "cost-mode": "array"
  },
  "endpoint-flows": [
    {
      "srcs": [ "ipv4:137.138.0.101" ],
      "dsts": [ "ipv4:134.158.84.23", "ipv4:144.16.112.112" ]
    },
    {
      "srcs": [ "ipv4:192.16.166.254" ],
      "dsts": [ "ipv4:140.115.32.101" ]
    }
  ],
  "ane-property-names": [ "next_hop", "as_path" ]
}
```

```
→curl -s -H 'Content-Type: application/alto-endpointcostparams+json' --data-ascii @
request-cern.json https://science.jensen-zhang.site/pathvector/cern-pv | ./pprint
--d41d8cd98f00b204e9800998ecf8427e
Content-Type: application/alto-endpointcost+json
Content-ID: <ecs@science.jensen-zhang.site>

{'endpoint-cost-map': {'137.138.0.101': {'134.158.84.23': ['autolink_1',
                                                           'autopath_2'],
                                         '144.16.112.112': ['autolink_1',
                                                            'autopath_3']},
                       '192.16.166.254': {'140.115.32.101': ['autolink_1',
                                                             'autopath_1']}},
 'meta': {'cost-type': {'cost-metric': 'ane-path', 'cost-mode': 'array'},
          'vtag': {'resource-id': 'cern-pv.ecs',
                   'tag': 'e615bf984f7249949f8903c5cf56f02d'}}}
--d41d8cd98f00b204e9800998ecf8427e
Content-Type: application/alto-propmap+json
Content-ID: <propmap@science.jensen-zhang.site>

{'meta': {'dependent-vtags': [{'resource-id': 'cern-pv.ecs',
                               'tag': 'e615bf984f7249949f8903c5cf56f02d'}]},
 'property-map': {'.ane:autolink_1': {'next_hop': '192.65.184.145'},
                  '.ane:autopath_1': {'as_path': '20965 24167 7539 1659'},
                  '.ane:autopath_2': {'as_path': '20965 2091 789'},
                  '.ane:autopath_3': {'as_path': '20965 9885 55824'}}}
--d41d8cd98f00b204e9800998ecf8427e--
```

Routing Plane Retrieval (Looking Glass of CERN and GEANT)

Implementation

etc > {} lg-agent.json > ...

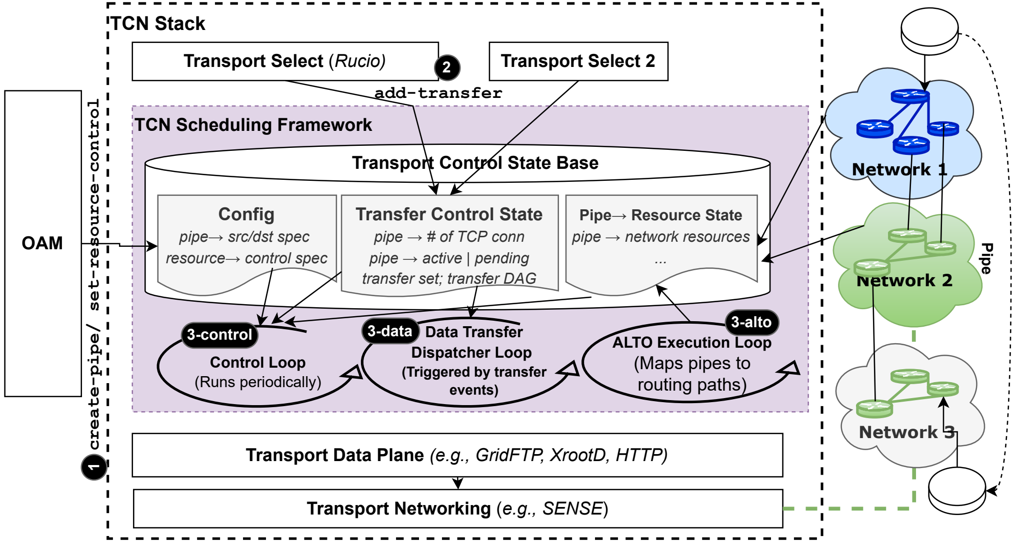
```
{
    "namespace": "default",
    "agent_class": "alto.agent.cernlg.LookingGlassAgent",
    "uri": "http://lhcone-lg.cern.ch/lg.cgi",
    "default_router": "ex2j.cern.ch:juniper",
    "refresh_interval": 300
}
```

Jensen/Kai/Lauren

Implementation: Control Loop

Integration into FTS 3.12

- Extend database schema for pipes (`t_link_config`) to support resource control specification (`tcn_abs_limit`, `tcn_rel_weight`)
- Implement `ALTO/TCNOptimizer` class for ALTO/FTS control loop
  - Implementing ZeroOrder Gradient with Integral, Quadratic Distance function
  - Add new optimizer mode (`kOptimizerAggregated`) to enable ALTO/FTS optimizer



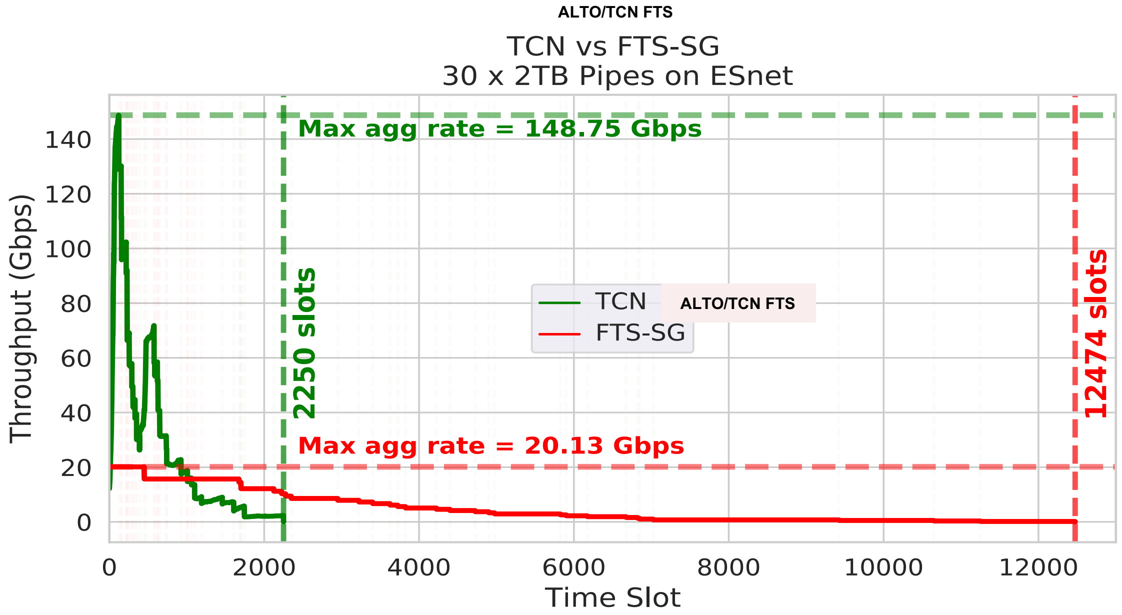
Milestones and Main Remaining Tasks

- Milestone
  - Wrap up implementation and test in summer 2023

- Some focusing tasks
  - Finalize language to specify resource model
  - Work with sites and higher-level workflows (e.g., Rucio) to specify typical control goals
  - Finish initial design of multi FTS instance control coordination
  - Integrate with infrastructure control (e.g., NOTED)

Backup Slides

Basic ALTO/FTS Benchmarking ⇒ Real Topology (ESnet)



Setting: 30 <src, dst> pipes, one request per pipe, each request 20K transfers, file size = 100MB. the total in the workload is 60TB.
Resource Control goal: all equal weights

1. **7.39x** total BW utilization.
2. **5.54x** Max RCT improvement. (Short-tailed)

Global Objective, Zero-order gradients, and Resource Control Constraints.