NeoTec
Internet-Draft
Intended status: Informational
Expires: 27 August 2026

L. Dunbar, Ed.
Futurewei
Q. Sun
China Telecom
B. Wu, Ed.
Huawei
L. CONTRERASMURILLO, Ed.
Telefornica
C. Xie
China Telecom

23 February 2026

Applying Attachmet Circuit and PE2PE YANG Data Model to dynamic policies Use Case
draft-dunbar-onsen-ac-pe2pe-ucmp-applicability-00

Abstract

This document explores how existing IETF YANG data models can be applied to support a use case involving dynamic, time-scoped UCMP (Unequal Cost Multipath) policy enforcement across multiple network segments interconnecting Edge Cloud sites. The use case is motivated by periodic, high-volume data exchanges between distributed AI inference modules placed at geographically dispersed edge data centers. By mapping network requirements such as bandwidth, latency, and availability to relevant YANG models, including AC, TE Topology, SR Policy, and QoS models, this document serves as a practical exercise to evaluate the applicability and limitations of current IETF specifications. It highlights the need for cloud-initiated, time-bounded network policy activation and identifies potential gaps in model expressiveness, policy lifecycle handling, and API-level abstraction required for real-time cloud-network coordination.

This Internet-Draft will expire on 27 August 2026.

Table of Contents

1. Introduction

Edge computing enables latency-sensitive applications, such as AI inference, to be deployed closer to end users and data sources. In many deployments, those latency sensitive applications are dynamically instantiated across multiple Edge Cloud sites based on compute availability, proximity to data sources, and overall service objectives. These distributed modules, especially AI inference instances, often need to exchange large volumes of data periodically, for example, during model synchronization, aggregated result sharing, or collaborative analytics.

These inter-site data exchanges typically require high bandwidth and low latency for short, well-defined time windows. However, most transport networks are optimized for long-lived, static flows and do not natively support time-scoped policy enforcement. Furthermore, existing routing mechanisms like Equal Cost Multipath (ECMP) are insufficient for granular traffic distribution when link costs and available bandwidth are unequal. As a result, Unequal Cost Multipath (UCMP) techniques have emerged to enable more efficient load balancing across heterogeneous paths.

This document focuses on the application of dynamic, cloud-initiated UCMP policy updates across multiple segments of the network interconnecting Edge Cloud sites. It assumes the network supports multiple paths between sites and that these paths can be selected or weighted based on real-time performance metrics (e.g., latency, available bandwidth, load).

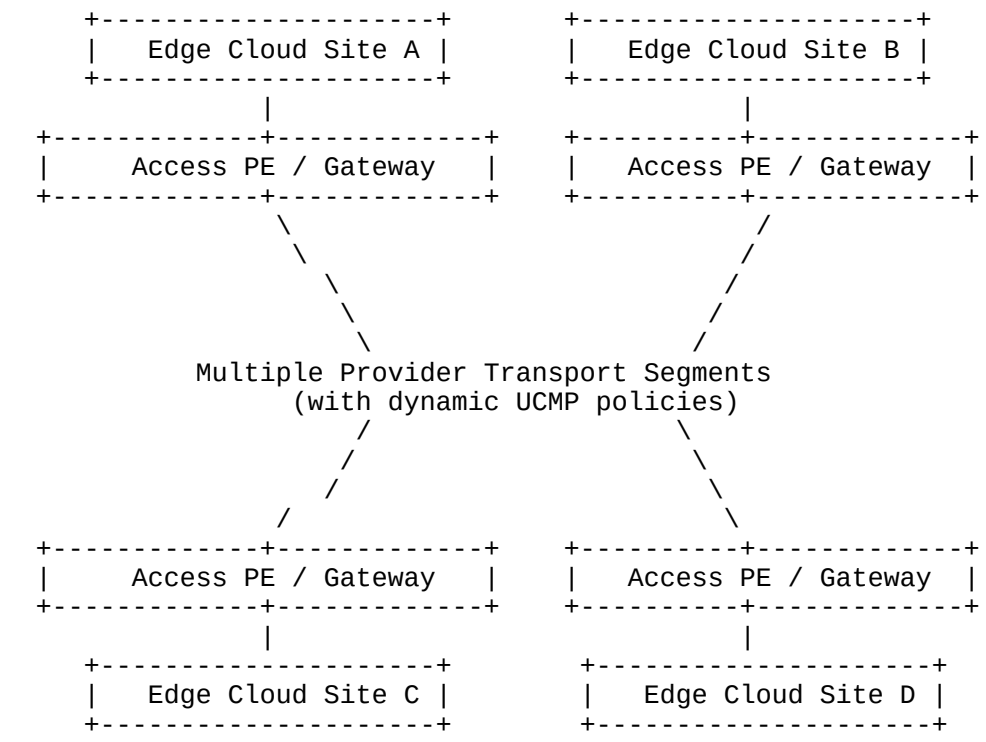

Figure 1: Network Segments for AI Data Exchange

The scenario described above can also be considered from a service abstraction perspective. The periodic inter-site AI synchronization represents a higher-layer service characterized by parameters such as minimum bandwidth, maximum latency, and a bounded activation interval. From the perspective of a Cloud Manager, the requirement may be expressed as a connectivity intent (e.g., provide a specified bandwidth between selected Edge Cloud sites within a defined time window) rather than as a specific routing or traffic engineering mechanism.

To realize such a service abstraction, a Network Orchestrator decomposes the intent into multiple network-level constructs. These may include routing context selection, traffic classification and QoS treatment, TE path selection, and weighted traffic distribution. Existing IETF YANG data models (e.g., ietf-routing, ietf-ac, ietf-te-topology, ietf-qos-policy, and ietf-sr-policy) provide configuration

primitives for these functions. However, these models operate largely independently and do not collectively define a unified, lifecycle-aware service abstraction.

This use case therefore illustrates an operational consideration: a time-scoped service objective may require coordinated updates across multiple models, network segments, and potentially administrative domains. Lifecycle handling (e.g., activation time, duration, and reversion behavior), policy interaction, and verification of service compliance are typically implemented in orchestration systems rather than expressed directly within the YANG models themselves. The UCMP examples in this document are used to explore this broader applicability and to identify areas where additional abstraction or coordination mechanisms may be beneficial.

The primary goal of this work is to:

- Demonstrate how IETF defined YANG data models, such as ietf-routing, ietf-ac, ietf-te-topology, ietf-qos-policy, and ietf-sr-policy, can be used to realize dynamic UCMP enforcemen.

- Highlight the requirements for time-scoped policy activation and cloud-initiated triggers, which are not natively supported in existing models.

- Propose augmentations or extensions needed to support temporary policy enforcement tied to cloud workload events.

- Identify architectural and modeling gaps that must be addressed to enable closed-loop coordination between cloud orchestration systems (e.g., Kubernetes) and network controllers.

By framing this scenario as a concrete use case, the document serves both as an applicability exercise and as input for future standardization efforts within the IETF aimed at cloud-network integration and SLA-driven policy control.

Disclaimer

The use of specific YANG data models (e.g., Attachment Circuit and TE topology) in this section is intended as a provisional exercise to explore how existing IETF models might address aspects of such a use case. These examples are not exclusive or exhaustive. Other data models, such as Network Slicing Service Model (NSSM) or service function chaining models, could also be relevant depending on the network architecture and service requirements. The intent is to assess the applicability and identify gaps (if any), not to pre-define the final solution set.

2. Conventions used in this document

Cloud Manager: An entity that is primarily responsible for placing workloads, managing compute resources across Edge Cloud sites, Monitoring the health and scaling status of VMs, containers, or services, Making application-level decisions (e.g., where to place a function based on latency, CPU, GPU availability), etc..

Network Orchestrator (or Orchestrator): A logical entity that interfaces with the Cloud Manager to receive service requests or queries and coordinates the end-to-end connectivity across multiple network domains. It abstracts underlying domain-specific technologies (e.g., L2/L3 VPNs, SR paths, TE tunnels) and disseminates policies to individual Network Controllers, enabling seamless stitching of diverse network segments to meet service-level requirements.

Network Controller: A domain specific control entity responsible for managing and configuring network resources within a single administrative or technological domain (e.g., IP/MPLS core, access network). It receives high-level intent or service instructions from a Network Orchestrator and translates them into device-level configurations using protocols such as NETCONF, BGP, or PCEP.

UE: User Equipment

UPF: User Plane Function [TS.23.501-3GPP]

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Dynamic UCMP Load Balancing for Periodic Inter Site AI Traffic

In the Neotec use case described in Section 1, AI inference modules are deployed across four Edge Cloud sites to support distributed city surveillance. These modules periodically exchange large volumes of data, for instance, during result aggregation or synchronized event analysis. These data exchanges are not continuous but are periodic and event driven, requiring guaranteed bandwidth and low latency for short time windows.

The underlying network connecting these Edge Cloud sites typically includes multiple paths between nodes and across multiple network segments.An end-to-end path between Edge Cloud Site A and B spans at least three segments:

- The first is the access segment from the Edge Cloud A to its closest PE. There could be multipe PEs to Edge Cloud A for multi-homing.

- The second segment traverses the provider transport network between the PE serving Edge Cloud A and the PE serving the Edge Cloud B.

- The third segment connects that Edge PE to the Edge Cloud B’s Gateway

3.1. UCMP Enforcement for Access Segment

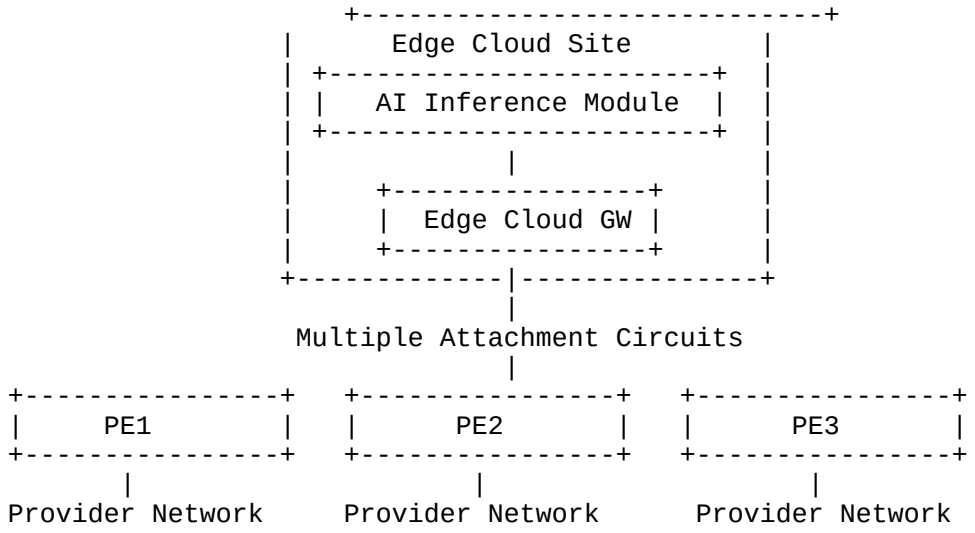

Figure 2: Edge Cloud Access Segment

The Edge Cloud Gateway has multiple logical links (attachment circuits) to a set of PEs (e.g., PE1, PE2, PE3). These links may vary in latency, bandwidth, or current load. During periodic AI data bursts, the Edge Cloud GW must push all other traffic away from PE1, PE2, and PE3, reserving their full available bandwidth for AI flows. Or it could push all other traffic to one of the PEs that has the lowest bandwidth and highest latency. This can be implemented by dynamically updating forwarding policies or QoS profiles to deprioritize or reroute non-AI traffic, ensuring that the AI inference module has uncontested access to network capacity during critical synchronization windows. Depending on the request from the Cloud Manager, the Network Orchestrator can determine what exact policies to push to the PEs and the Edge GW..

YANG Data Models for Policy Enforcement:

To support dynamic UCMP-based traffic steering across PE1, PE2, and PE3, the Network Orchestrator can utilize the following IETF YANG models:

- ietf-routing [RFC8349] enables configuration of routing instances and static routes. The data model allows per-prefix route entries with multiple weighted next hops, supporting unequal cost paths. It can be used to define static next-hop policies from the Edge GW toward PE1, PE2, and PE3.

- ietf-qos-policy [I-D.claise-opsawg-qos-packet-marking] defines traffic classification, marking, and treatment policies. It can enforce rate limits or scheduling priority for non-AI traffic routed to PE3.

- ietf-ac [I-D.ietf-opsawg-teas-attachment-circuit] can provides performance characteristics (bandwidth, latency, packet loss), which inform the decision to assign traffic classes.

- ietf-traffic-classifier / ietf-packet-policy (or OpenConfig equivalents) can be used to match traffic classes (e.g., AI vs non-AI flows), enables forwarding decisions at the Edge GW and ingress of PEs.

Policy Example: Prioritized AI Flow Assignment and Non-AI Rerouting:

Assuming PE1 has the highest available capacity and best latency toward the remote site:

The orchestrator uses ietf-routing [RFC8349] to define weighted static next hops at the Edge GW:

- PE1: 2/3 of AI traffic.

- PE2: 1/3 of AI traffic.

- PE3: no AI traffic.

```
{
  "routing:routing": {
    "routing-instance": [
      {
        "name": "ai-routing-instance",
        "routing-protocols": {
          "routing-protocol": [
            {
              "type": "static",
              "name": "ai-policy",
              "static-routes": {
                "ipv4": {
                  "route": [
                  {
                    "destination-prefix": "198.51.100.0/24",
                    "next-hop": [
                    { "outgoing-interface": "to-PE1", "weight": 66},
                    { "outgoing-interface": "to-PE2", "weight": 34}
                    ]
                  }
                  ]
                }
              }
            }
          ]
        }
      }
    ]
  }
}
```

Figure 3: UCMP policy example

Non-AI traffic is matched by classifiers and redirected entirely to PE3:

- Apply QoS policy using ietf-qos-policy [I-D.claise-opsawg-qos-packet-marking] to limit available bandwidth on PE3 to non-AI traffic.

- PE3 may be tagged "degraded" in the ietf-ac model to signal its backup nature.

This ensures that the AI traffic is prioritized through the most capable paths (PE1 and PE2) with precise weight. All non-critical traffic is offloaded to the least-desired path (PE3), reducing congestion. Orchestrator needs to dynamically adapt this logic per service request.

Fallback Logic: If available capacity on PE1 and PE2 is not sufficient, AI traffic still receives best possible routing via PE1 and PE2, and Non-AI traffic may be allowed limited access to PE2 with reduced weight (e.g., 10%) while still primarily routed via PE3.

This tiered policy enforcement ensures strict adherence to SLA goals for AI inference while maintaining service continuity for other traffic classes using standard YANG-based configuration interfaces.

One critical limitation in existing IETF YANG models is the lack of native support for time-scoped UCMP policy activation. To support the bursty nature of AI traffic, the following strategies can be applied:

- Define policy activation via external triggers from the Cloud Manager using an API call to the orchestrator.

- Orchestrator maintains a mapping between the requested activation time window and the temporary configuration to be applied.

- Extend existing YANG models (e.g., ietf-routing or ietf-sr-policy) with optional augmentation for: start-time, duration, and expiration-action

Example augmentation (conceptual):

```
augment "/routing:routing/routing-instance/static-routes/route" {
  leaf burst-policy-start-time {
    type yang:date-and-time;
  }
  leaf burst-policy-duration-sec {
    type uint32;
  }
  leaf expiration-action {
    type enumeration {
      enum revert-to-default;
      enum retain;
    }
  }
}
```

Figure 4: Example Augmentation

In the absence of standard YANG support, this behavior need to be implemented in the orchestrator logic by maintaining policy lifecycle state and timers, pushing temporary configuration via NETCONF/ RESTCONF at start-time, and reverting after duration-sec.

3.2. UCMP Enforcement for SRv6 based PE to PE segment

Assuming an SRv6 underlay among the PEs, the network controller can use the ietf-sr-policy YANG model to update the traffic distribution weights across pre-established paths. For example, if three SRv6 paths exist between EdgeSite-A and EdgeSite-C, the controller can push the following configuration to the ingress node:

```
sr-policy {
  color 4001;
  endpoint "2001:db8:100::1";
  candidate-paths {
    preference 100;
    path {
      weight 70;
      sid-list [2001:db8:10::1, 2001:db8:11::2];
    }
    path {
      weight 20;
      sid-list [2001:db8:20::1, 2001:db8:21::2];
    }
    path {
      weight 10;
      sid-list [2001:db8:30::1, 2001:db8:31::2];
    }
  }
}
```

Figure 5: Using SR Policy

This UCMP configuration tells the network to distribute traffic unequally across the three paths based on their capability. The underlying topology and metrics are derived from ietf-TE-topology and ietf-TE models, which expose bandwidth, latency, and available resources for each link.

Similar UCMP behavior can also be implemented over SR-MPLS, MPLS-TE, or enhanced IP networks, using the corresponding IETF YANG models (ietf-TE, ietf-routing, etc.). The key point is that the network paths are preexisting, and the only dynamic action is adjusting how traffic is forwarded among them in response to a cloud service request.

3.3. UCMP Enforcement for PE to PE segment in Non-SRv6 networks

In many operator networks that do not support SRv6, the PE-to-PE segment is realized as a logical transport tunnel (e.g., MPLS-TE LSP, IPsec tunnel, or GRE) that may traverse multiple intermediate routers. Each of these routers may have multiple outgoing links to their respective next hops, making end-to-end UCMP behavior dependent on both tunnel-level path selection and interior node forwarding decisions.

Policy Application for UCMP in Non-SRv6 Networks:

To enable UCMP behavior between PEs in non-SRv6 networks, the network controller can provision multiple TE tunnels or IP-layer paths with distinct performance characteristics. The UCMP forwarding behavior is realized by adjusting traffic distribution weights among these logical tunnels at the ingress PE (or head-end router). This setup also requires the orchestrator to be topology-aware and TE-capable.

3.3.1. UCMP over MPLS-TE

In MPLS-TE-enabled networks, PE-to-PE traffic is carried over logical LSPs that are pre-established and maintained by the network controller. These LSPs can be engineered with specific bandwidth, latency, or disjointness constraints and serve as the building blocks for UCMP.

Key Characteristics:

- UCMP logic is enforced at the ingress PE (head-end LSR)

- Intermediate routers simply perform label switching and do not require knowledge of traffic distribution policies.

- The network controller selects multiple LSPs and configures the PE to assign weighted traffic across them.

YANG Models Used

- ietf-te-topology [RFC8795]: Describes available TE links and nodes.

- ietf-routing [RFC8349]: Configures static routes with weighted next-hops over the tunnels.

```
      {
  "routing:routing": {
    "routing-instance": [{
      "name": "ucmp-te",
      "routing-protocols": {
        "routing-protocol": [{
          "type": "static",
          "name": "te-policy",
          "static-routes": {
            "ipv4": {
              "route": [{
                "destination-prefix": "198.51.100.0/24",
                "next-hop": [
               {"outgoing-interface": "mpls-te-tunnel-A", "weight": 60},
               {"outgoing-interface": "mpls-te-tunnel-B", "weight": 40}
               ]
             }]
            }
          }
        }]
      }
    }]
  }
}
```

Figure 6: UCMP over MPLS-TE example

In this model, the network controller dynamically adjusts the route weights during AI synchronization periods based on the request from the service orchestrator, favoring higher-performing tunnels.

3.3.2. UCMP Over Plain IP (ECMP)

In networks without MPLS, traffic between PEs may traverse IP-based ECMP paths. Each router independently forwards traffic using its own hash-based load balancing over equal-cost links. Enabling UCMP over such paths is more complex because intermediate routers may not natively support weighted forwarding.

Options for UCMP Enforcement:

- Head-End-Based WCMP (Weighted ECMP): If routers support WCMP extensions, the ingress PE can apply traffic weights across next-hops. The rest of the network performs standard ECMP; however, overall path selection remains coarse.

- Hop-by-Hop Configuration: If consistent UCMP behavior is required, each router along the path must be configured to support either: Weighted static routes using ietf-routing, or Traffic classifiers and filters using ietf-traffic-classifier or OpenConfig equivalents. This approach increases operational complexity and may require additional non-trival control logic.

YANG Models Used:

- ietf-routing [RFC8349]: Configures static or policy-based routes with weighted next-hops.

- ietf-qos-policy [I-D. claise-opsawg-qos-packet-marking] and ietf-traffic-classifier [I-D. opsawg-traffic-classifier]: Used for traffic tagging and class-based forwarding.

Limitations: many routers do not support WCMP or allow fine-grained weight control for ECMP paths. Lack of coordination across routers can lead to inconsistent forwarding behavior. Telemetry feedback from mid-path nodes is often unavailable or vendor-specific.

Here is a JSON-based example using IETF YANG models to enforce UCMP behavior via hop-by-hop configuration, assuming each router along the path supports: Weighted static routes (ietf-routing), Traffic classification (ietf-traffic-classifier), and QoS forwarding rules (ietf-qos-policy).

This example configures each router to: Match AI-related traffic using a classifier, Apply weighted next-hops via ietf-routing, and Enforce treatment policies using QoS configuration.

1. Traffic Classifier (per router)

```
{
  "traffic-classifier:classifiers": {
    "classifier": [
      {
        "name": "ai-traffic",
        "description": "Classify traffic for AI application",
        "match": {
          "ipv4": {
            "destination-address": "198.51.100.0/24"
          },
          "dscp": 34  // Assumes AI flows are marked with DSCP AF41
        }
      }
    ]
  }
```

```
 }
```

2. QoS Policy for Forwarding Behavior:

```
 {
  "qos-policy:policies": {
    "policy": [
      {
        "name": "ucmp-ai-policy",
        "classifier": "ai-traffic",
        "forwarding-actions": {
          "outgoing-interfaces": [
            { "interface": "eth1", "weight": 70 },
            { "interface": "eth2", "weight": 30 }
          ]
        }
      }
    ]
  }
 }
```

3. Weighted Static Route for UCMP (applied per-hop)

```
{
  "routing:routing": {
    "routing-instance": [
      {
        "name": "ucmp-instance",
        "routing-protocols": {
          "routing-protocol": [
            {
              "type": "static",
              "name": "static-ucmp",
              "static-routes": {
                "ipv4": {
                  "route": [
                    {
                      "destination-prefix": "198.51.100.0/24",
                      "next-hop": [
                        { "outgoing-interface": "eth1", "weight": 70 },
                        { "outgoing-interface": "eth2", "weight": 30 }
                      ]
                    }
                  ]
                }
              }
            }
          ]
        }
      }
```

```
    ]
  }
}
```

Figure 7: Hop by Hop UCMP example

This configuration must be replicated (with appropriate interface changes) on each router along the PE-to-PE path.Traffic classification ensures only AI flows are subject to UCMP behavior.Each router’s forwarding engine must support policy-based routing or weighted ECMP for this to work.Orchestration systems must maintain synchronized state and rollback mechanisms across devices.

4. Cloud-Initiated UCMP Activation API

A simplified example of a cloud-initiated API call to the network controller might look like:

```
POST /network-policy/ucmp-activation
{
  "source-sites": ["EdgeSite-A", "EdgeSite-B"],
  "dest-sites": ["EdgeSite-C", "EdgeSite-D"],
  "start-time": "2025-05-01T10:00:00Z",
  "duration-sec": 300,
  "min-bandwidth-mbps": 5000,
  "max-latency-ms": 10
}
```

Figure 8: Burst Network Request

This request informs the network controller that a high-volume, low-latency data exchange will occur and that UCMP forwarding policies should be applied to optimize transport between the specified sites for the specified duration.

5. Gaps Analysis

This section evaluates the limitations and shortcomings of current IETF YANG models and network orchestration mechanisms in supporting dynamic, time-scoped UCMP policy enforcement across both Access and PE-to-PE segments in multi-segment networks interconnecting Edge Cloud sites.

5.1. Access Segment Gaps Analysis

In the access segment between Edge Cloud gateways and PEs (e.g., PE1, PE2, PE3), traffic distribution policies can be enforced using weighted static routing (ietf-routing) and QoS classifiers (ietf-qos-policy). However, the following issues remain:

- Lack of Time-Bound Policy Support: Existing YANG models do not support time-bound routing or QoS policies natively. Implementing time-scoped UCMP (e.g., apply for 5 minutes during AI sync) requires custom orchestrator logic, including timers, state tracking, and reversion mechanisms.

- No Native Lifecycle Hooks: YANG models like ietf-ac and ietf-routing lack fields for lifecycle hooks such as start-time, duration, or expiration-behavior, making transient policy management cumbersome.

- Granular Traffic Steering Limitations: While attachment circuit metrics can inform traffic placement decisions, there is no standardized way to associate UCMP weights directly with circuit performance in an event-driven manner.

- Policy Interference Risk: There is no clear conflict resolution mechanism when temporary UCMP policies override existing long-term configurations. Operators may hesitate to automate access path reconfigurations due to unpredictable side effects.

5.2. PE to PE Segment Gaps Analysis

For the transport segment between PEs (especially non-SRv6 networks), dynamic UCMP faces additional modeling and orchestration challenges:

- Inconsistent UCMP Support Across Technologies: While SRv6 supports weighted traffic steering via ietf-sr-policy, non-SRv6 networks (e.g., MPLS-TE, SR-MPLS, or IP) lack consistent models for applying and adjusting UCMP behavior dynamically. The UCMP logic often has to be emulated via unequal TE tunnel provisioning and static route weight tuning.

- Topology Model Limitations: ietf-te-topology and ietf-te expose rich metrics but are not inherently reactive to cloud events. There is no trigger mechanism to automatically instantiate or adjust TE tunnels based on time-scoped cloud service requests.

- Limited Feedback Loops: Current YANG models are largely configuration-driven. There is no standardized feedback channel from devices to orchestrators to confirm UCMP policy status, enforcement success, or SLA compliance during active windows.

5.3. Complexity of Hop-by-Hop UCMP Configuration in IP Networks

In IP-based networks without MPLS-TE or SRv6, traffic distribution across multiple paths typically relies on Equal Cost Multipath (ECMP), where each router independently forwards traffic based on local hash-based algorithms. Applying Unequal Cost Multipath (UCMP) in such environments requires hop-by-hop configuration to ensure consistent behavior, introducing substantial complexity.

Identified Gaps

- Lack of Unified UCMP Capability Discovery: Current IETF YANG models (ietf-routing, ietf-interfaces) do not expose whether a router supports Weighted ECMP (WCMP), the granularity of supported weights, or the hashing algorithm used for multipath forwarding. Network controller cannot dynamically determine whether a node can participate in UCMP enforcement.

- No Model for Coordinated Hop-by-Hop Weight Distribution:There is no standardized method to propagate UCMP weights across multiple nodes consistently. ietf-routing supports static routes with weighted next-hops, but each router must be configured independently with no assurance that downstream nodes share the same view.

- Limited Telemetry for Flow Distribution Verification: Existing telemetry models (e.g., ietf-interfaces, ietf-te) offer counters per interface, but do not expose real-time per-class or per-prefix distribution across ECMP paths. Operators lack visibility into whether UCMP objectives are being met without custom probes or vendor-specific tools.

- Policy Lifecycle and Rollback Limitations: Time-scoped UCMP policies require precise activation and rollback behavior. Current models lack lifecycle attributes (e.g., start time, expiration behavior) to automate the temporal aspect of UCMP enforcement.

5.4. Inter-Domain Coordination Gaps Analysis

- Cross-Domain Policy Stitching: When the PE-to-PE path traverses multiple administrative domains, coordinating UCMP policy enforcement becomes challenging. There is no standard way for an orchestrator to request and verify consistent path weights across AS boundaries.

- Cloud-to-Network Intent Translation: Cloud controllers (e.g., Kubernetes) cannot directly express intent such as "min 5Gbps to EdgeSite-C from 10:00-10:05 UTC" in a format consumable by TE or routing models. This necessitates the creation of a new intent or policy model aligned with both cloud-native and IETF constructs.

6. Security Considerations

To be added

7. IANA Considerations

None

8. References

8.1. Normative References

8.2. Informative References

[Neotec-Zerotrust-Access]
L. Dunbar and H. Chihi, "Neotec-Zerotrust-Access", December 2024, <https://datatracker.ietf.org/doc/draft-dunchihi-neotec-zerotrust-access-dm/>.

[opsawg-teas-attachment-circuit]
M. Boucadair, et al, "opsawg-teas-attachment-circuit", January 2025, <https://datatracker.ietf.org/doc/draft-ietf-opsawg-teas-attachment-circuit/>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8345] Clemm, A., Medved, J., Varga, R., Bahadur, N., Ananthakrishnan, H., and X. Liu, "A YANG Data Model for Network Topologies", RFC 8345, DOI 10.17487/RFC8345, March 2018, <https://www.rfc-editor.org/info/rfc8345>.

[RFC8639] Voit, E., Clemm, A., Gonzalez Prieto, A., Nilsen-Nygaard, E., and A. Tripathy, "Subscription to YANG Notifications", RFC 8639, DOI 10.17487/RFC8639, September 2019, <https://www.rfc-editor.org/info/rfc8639>.

[RFC8641] Clemm, A. and E. Voit, "Subscription to YANG Notifications for Datastore Updates", RFC 8641, DOI 10.17487/RFC8641, September 2019, <https://www.rfc-editor.org/info/rfc8641>.

[RFC8776] Saad, T., Gandhi, R., Liu, X., Beeram, V., and I. Bryskin, "Common YANG Data Types for Traffic Engineering", RFC 8776, DOI 10.17487/RFC8776, June 2020, <https://www.rfc-editor.org/info/rfc8776>.

[RFC8795] Liu, X., Bryskin, I., Beeram, V., Saad, T., Shah, H., and O. Gonzalez de Dios, "YANG Data Model for Traffic Engineering (TE) Topologies", RFC 8795, DOI 10.17487/RFC8795, August 2020, <https://www.rfc-editor.org/info/rfc8795>.

[TS.23.501-3GPP] 3rd Generation Partnership Project (3GPP), "System Architecture for 5G System; Stage 2, 3GPP TS 23.501 v2.0.1", December 2017.

Acknowledgements

The authors would like to thank for following for discussions and providing input to this document: xxx.

Contributors

Authors' Addresses

Linda Dunbar (editor)
Futurewei
United States of America
Email: ldunbar@futurewei.com

Qiong
China Telecom
China
Email: sunqiong@chinatelecom.cn

Wu Bo (editor)
Huawei
China
Email: lana.wubo@huawei.com

LUIS MIGUEL CONTRERAS MURILLO (editor)
Telefornica
Spain
Email: luismiguel.contrerasmurillo@telefonica.com

ChongFeng Xie
China Telecom
China
Email: xiechf@chinatelecom.cn