(ICC 2021)

Optimal Multicast Service Chain Control: Packet Processing, Routing, Duplication

Yang Cai, Jaime Llorca, Antonia M. Tulino, Andreas F. Molisch

Email: yangcai@usc.edu

Background

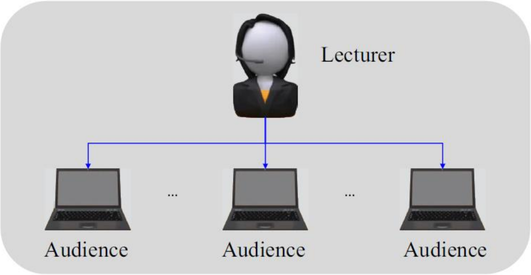
- Increasing demand for computational resource
  - Augmented reality, telepresence, industrial automation
- Limited local computational resource at UE
  - Tendency: light weight, portable devices
  - Restricted processing capability, battery
- Solution: offloading computing tasks to the cloud
  - Better delay and cost performance

Background

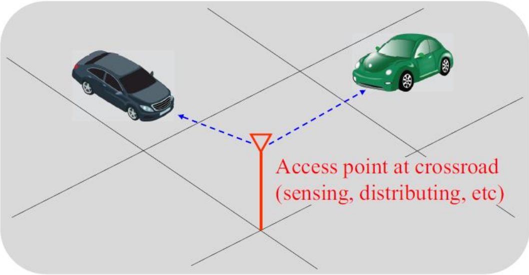
- Distributed cloud network
  - Make it easier for the UEs to access the computational resource
    - Traditional processing network: separation of network & processing center
    - Distributed cloud network: deploy the computational resource in a more widespread manner
- NFV & SDN-enabled Next-Gen Cloud
  - Make it more flexible for the cloud to process the data-stream
    - Augmented information service: computing task → service function chain
    - Each individual function can be implemented separately (at different network locations)

Background

- Multicast flow is an increasingly dominant component of the network traffic
  - Applications with multiple destinations
    - Multi-user conferencing

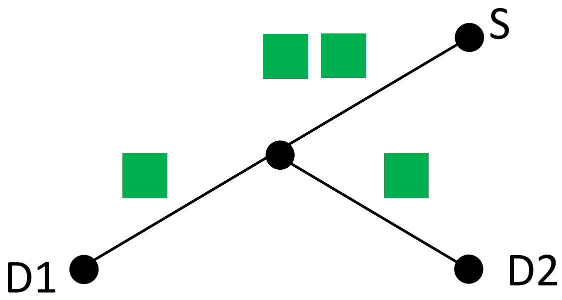


  - Coordinated decision making/content distribution within multi-agent systems
    - Intelligent transportation system, smart factory

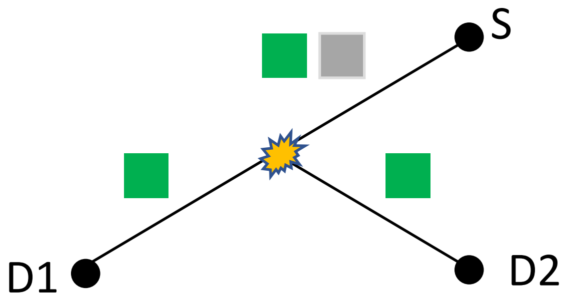


Background

- Existing Studies
  - Heuristic algorithms
    - Unicast approach: treat multicast flow as several individual unicast flows



Unicast approach



New scheme with flexible duplication

  - Throughput-optimal, least cost multicast routing
    - Few results (centralized control requiring global information)

System Model

- Cloud network model
  - Node: data-centers
    - Computation capacity C_i, associated cost e_i
  - Edge: transmission link
    - Transmission capacity C_{ij}, associated cost e_{ij}

System Model

- Service model
  - Service function chain: a chain of ordered functions
    - Two parameters: scaling factor $\xi_\phi^{(m)}$, workload $r_\phi^{(m)}$
  - Destination set $\mathcal{D}$ (multicast)
    - It becomes a singleton $\mathcal{D} = \{d\}$ for unicast flow
  - I.I.D. requests from each client
    - $a_i(t)$: the instantaneous arrival
    - λ_i : mean arrival rate

System Model

- Duplication operation
  - Current destination set
    - Keep track of the destination set of each copy of a content
  - Duplication status vector
    - Binary vector
    - Example: $q = (0, 1)$, the service has two destinations, and the packet is assigned the second destination

System Model

- Duplication operation (assumptions)
  - The *coverage* constraint

$$\underbrace{q}_{\text{original packet}} \leq \underbrace{s + r}_{\text{two copies}}$$

    - *Efficient* duplication: destination sets of the copies do not overlap

$$q = s + r$$

  - 2-duplication

Policy Space

- Policy space
  - Decision variable: the flow variables $\boldsymbol{f}(t) = \{f_{i,\mathrm{pr}}^{(q)}(t), f_{ij}^{(q)}(t)\}$
  - Constraints
    1. Non-negativity $\boldsymbol{f}(t) \succeq 0$
    2. Capability constraint

$$\boxed{\tilde{f}_i(t) = \sum_q r_\phi^{(m)} f_{i,\mathrm{pr}}^{(q)}(t) \le C_i}, \quad \boxed{f_{ij}(t) = \sum_q f_{ij}^{(q)}(t) \le C_{ij}}$$

    3. Generalized flow conservation law

Policy Space

- Policy space
  - Decision variable
  - Constraints

    3. Generalized flow conservation law (coverage constraint)

$$f^{(q)}_{i\to} = \overline{\left\{f^{(q)}_{i,\mathrm{pr}}(t) + \sum\nolimits_{j\in\delta_i^+} f^{(q)}_{ij}(t)\right\}} \text{ with } \overline{\{z(t)\}} = \lim_{T\to\infty}\frac{1}{T}\sum_{t=0}^{T-1} z(t)$$

$$\sum_{\boxed{\{q:q_k=1\}}} \left[f^{(q)}_{\to i} + \lambda^{(q)}_i\right] \le \sum_{\{q:q_k=1\}} \boxed{f^{(q)}_{i\to}}$$

Duplication status with destination d_k in it

Total incoming packets targeting d_k

Total outgoing packets targeting d_k

Policy Space

- Policy space
  - Decision variable
  - Constraints

3. Generalized flow conservation law (efficient duplication)

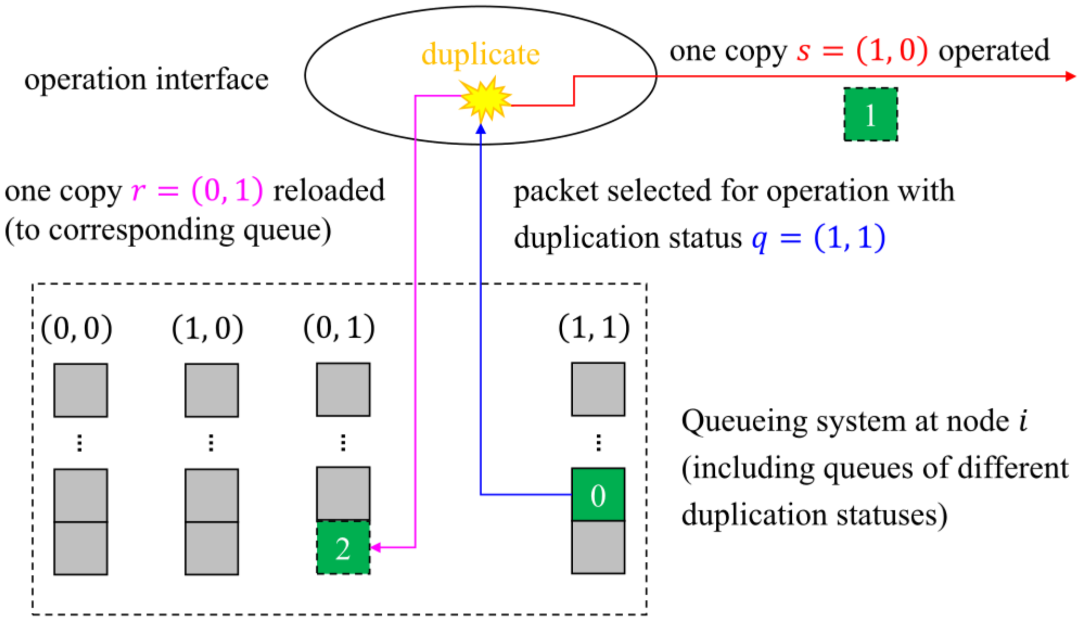
   (it does not reduce throughput, but can save cost and network traffic)

$$\sum_{\{q:q_k=1\}} \left[f_{\to i}^{(q)} + \lambda_i^{(q)}\right] \boxed{=} \sum_{\{q:q_k=1\}} f_{i\to}^{(q)}$$

Queuing System

- Queuing system
  - Queues $\boldsymbol{Q}(t) = [Q_i^{(q)}(t)]$
    - The queue commodity (c, q) at node i on time slot t
  - Flow variables $\boldsymbol{x}(t) = [x_{ij}^{(q,s)}(t)]$
    - The planned number of commodity q packets, such that they are duplicated into two copies with status s and $q - s$, with the first copy sent from node i to j, and the second copy reloaded
    - This variable specifies both the duplication and the routing decisions

Queuing System



- Queuing system
  - An example
  - Queuing dynamics
    - Intermediate nodes

$$Q_i^{(q)}(t+1) \le \left[Q_i^{(q)}(t) - \sum_{s \in 2^q} \mu_{i\to}^{(q,s)}(t) \right]^+ + \mu_{\to i}^{(q)}(t) + a_i^{(q)}(t)$$

$$\mu_{i\to}^{(q,s)}(t) = x_{i,\mathrm{pr}}^{(q,s)}(t) + \sum_{j \in \delta_i^+} x_{ij}^{(q,s)}(t)$$

$$\mu_{\to i}^{(q)}(t) = \boxed{\sum_{s \in 2^{\bar q}} \left[x_{\mathrm{pr},i}^{(q+s,q)}(t) + \sum_{j \in \delta_i^-} x_{ji}^{(q+s,q)}(t) \right]} + \boxed{\sum_{s \in 2^{\bar q}} \mu_{i\to}^{(q+s,s)}(t)}$$

Problem Formulation

$$\min_{\boldsymbol{x}(t)} \quad \overline{\{\mathbb{E}\{h(t)\}\}} \qquad h(t) = \boxed{\sum_{i \in \mathcal{V}} e_i \tilde{f}_i(t)} + \boxed{\sum_{(i,j) \in \mathcal{E}} e_{ij} f_{ij}(t)}$$

$$\text{s. t.} \quad \text{stabilize } \boldsymbol{Q}(t)$$

$$x_{\text{pr},i}^{(\phi,m+1,\mathcal{D},q,s)}(t) = \xi_\phi^{(m)} x_{i,\text{pr}}^{(\phi,m,\mathcal{D},q,s)}(t)$$

$$\tilde{x}_i(t) \triangleq \sum_{(c,q,s)} r_\phi^{(m)} x_{i,\text{pr}}^{(c,q,s)}(t) \le C_i \quad \forall\, i \in \mathcal{V}$$

$$x_{ij}(t) \triangleq \sum_{(c,q,s)} x_{ij}^{(c,q,s)}(t) \le C_{ij} \quad \forall\, (i,j) \in \mathcal{E}$$

$$\boldsymbol{x}(t) \succeq 0$$

Proposed Algorithm

- Based on Lyapunov Drift-plus-Penalty (LDP) approach
  - Goal: to minimize LDP given by

$$\Delta(t) + Vh(t) \leq B - \sum_{i \in \mathcal{V}} \sum_{(c,q,s)} \boxed{w_i^{(c,q,s)}} x_{i,\mathrm{pr}}^{(c,q,s)}(t) - \sum_{(i,j) \in \mathcal{E}} \sum_{(c,q,s)} \boxed{w_{ij}^{(c,q,s)}} x_{ij}^{(c,q,s)}(t)$$

$$\boxed{w_i^{(c,q,s)} = \frac{Q_i^{(c,q)}(t) - Q_i^{(c,q-s)}(t) - \xi_\phi^{(m)} Q_i^{(c',s)}(t)}{r_\phi^{(m)}} - Ve_i}$$

$$\boxed{w_{ij}^{(c,q,s)} = Q_i^{(c,q)}(t) - Q_i^{(c,q-s)}(t) - Q_j^{(c,s)}(t) - Ve_{ij}}$$

Proposed Algorithm

- Algorithm

1) calculate the weight for each tuple (c, q, s);
2) find the tuple (q, s, c) with the largest weight, i.e.,

$$(q,s,c)^\star = \arg\max_{(q,s,c)} \; w_i^{(q,s,c)} \; \left(\text{or } w_{ij}^{(q,s,c)}\right);$$

3) the optimal flow assignment is given by

$$x_{i,\text{pr}}^{(q,s,c)}(t) = \frac{C_i}{r_{\phi^\star}^{(m^\star)}} \, \mathbb{I}\left\{(q,s,c) == (q,s,c)^\star, w_{ij}^{(q,s,c)^\star}(t) > 0\right\}$$

$$x_{ij}^{(q,s,c)}(t) = C_{ij} \, \mathbb{I}\left\{(q,s,c) == (q,s,c)^\star, w_{ij}^{(q,s,c)^\star}(t) > 0\right\}$$

where $\mathbb{I}\{\cdot\}$ denotes the indicator function, which equals to 1 only when the two conditions are both satisfied.

Performance Analysis

- Delay-cost trade-off

Theorem 2: For any arrival vector $\boldsymbol{\lambda}$ that is in the interior of the capacity region, the queue backlog and the cost achieved by the proposed algorithm satisfy

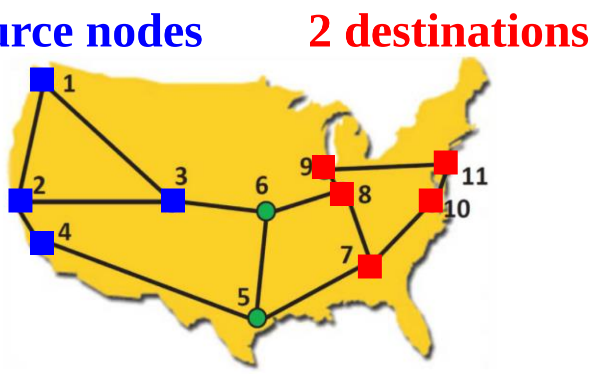
$$\overline{\{\mathbb{E}\{\|\boldsymbol{Q}(t)\|_1\}\}} \leq \frac{B}{\epsilon} + \left[\frac{h^\star(\boldsymbol{\lambda}+\epsilon\mathbf{1}) - h^\star(\boldsymbol{\lambda})}{\epsilon}\right] V \sim \mathcal{O}(V)$$

$$\overline{\{\mathbb{E}\{h(t)\}\}} \leq h^\star(\boldsymbol{\lambda}) + \frac{B}{V} \sim \mathcal{O}(1/V)$$

for any $\epsilon > 0$ such that $\boldsymbol{\lambda} + \epsilon\mathbf{1} \in \Lambda$.

  – The average delay is proportional to $\overline{\{\mathbb{E}\{\|\boldsymbol{Q}(t)\|_1\}\}} \sim \mathcal{O}(V)$

Numerical Experiments



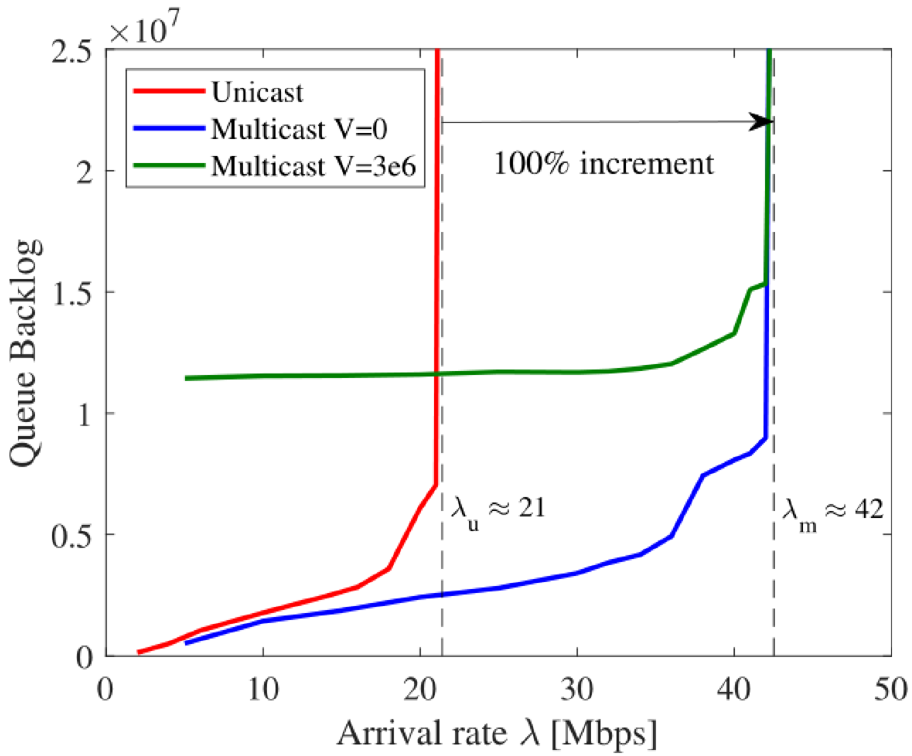
- Configuration
  - Network topology (Abilene network)
  - Available resource & cost
    - The computational resource is 20 CPUs at any node, with cost 0.5 /CPU
    - The transmission resource is 10 Gbps for any link, with a cost of 1 /Gb
  - Provided service
    - Two AgI services, each including two functions with parameter:

$$\phi_1 : \ \xi_1^{(1)} = 1, \ \xi_1^{(2)} = 2; \ 1/r_1^{(1)} = 300, \ 1/r_1^{(2)} = 400$$

$$\phi_2 : \ \xi_2^{(1)} = 1/3, \ \xi_2^{(2)} = 1/2; \ 1/r_2^{(1)} = 200, \ 1/r_2^{(2)} = 100$$

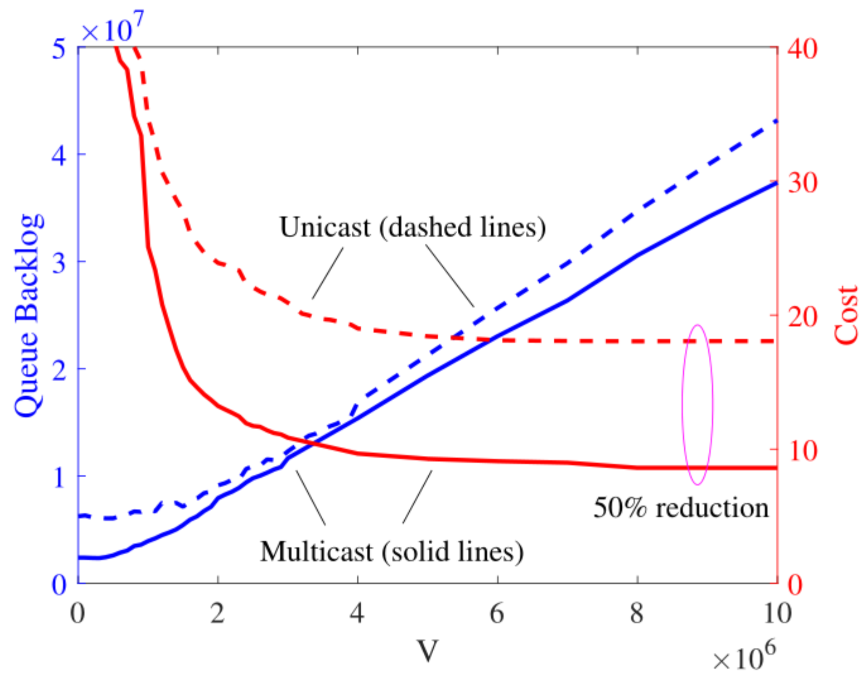
Numerical Experiments

- Network capacity region
  - Different V value leads to the same capacity region
  - Achieve $2 \times$ the capacity region of unicast approach
    - Since there are 2 destinations



Numerical Experiments

- Tradeoff controlled by V parameter
  - $[O(V),\ O(1/V)]$ tradeoff between queue backlog and the achieved cost
  - Achieve $1/2 \times$ the cost of unicast approach
    - Since there are 2 destinations



Conclusions

- The delivery of multicast network flow can be optimized by exploiting the reuse gain
- Efficient duplication can 1) ensure information delivery to all destinations, and 2) save the network resource
- The proposed approach can support the entire capacity region and achieve near-optimal cost performance

Acknowledgement

- Thanks for joining in the talk!
- Please contact yangcai@usc.edu if you have any questions, comments
- The most recent results on this topic (which deals with mixed-cast flows) are under preparation for submission to IEEE Trans. Comm.

Queuing System

- Queuing system
  - Queuing dynamics
    - Destination $i = d_k$

$$Q_i^{(c,q)}(t+1) \le \begin{cases} \boxed{0} & q_k = 1 \\ R + \boxed{\mu_{\to i}^{(c,q+b_k)}(t) + a_i^{(c,q+b_k)}(t)} & q_k = 0 \end{cases}$$

    - Example: what happens if a $(1,1)$ packets arrives at d_1

$(1,1)$

$(1,0)$: depart the network since it is at d_1

$(0,1)$: join the queue for $(0,1)$ at d_1

System Model

- Request model
  - Destination set $\mathcal{D}$ (multicast)
    - It becomes a singleton $\mathcal{D} = \{d\}$ for unicast flow
  - Commodity $c = (\mathcal{D}, \phi, m)$
    - (destination set, request service, current stage in the service function chain)
  - I.I.D. requests from each client
    - $a_i^{(c,q)}(t)$: the instantaneous arrival
    - $\lambda_i^{(c,q)}$: mean arrival rate