UNIVERSITY OF CAMBRIDGE

Digital Fountains in (our) ICN
Opportunities for Resource Management

George Parisis, Computer Laboratory

Introduction

- ICN is popular
  - *what* is more important than *who*
  - Information items can be labeled (on content, chunk, packet level)
  - Multicast is the norm
- Reliable information dissemination
- Resource management across the network

Contribution

- So far only pull-based mechanisms
  - End-to-content, hop-by-hop
- Can we use the power of separating core functions to design a resource management approach?
  - Better control over network resources..
  - Virtually centralized control points to manage network resources (controlled by entities that must control their resources)

Digital Fountains for Reliable Information Dissemination and Resource Management

Digital Fountains

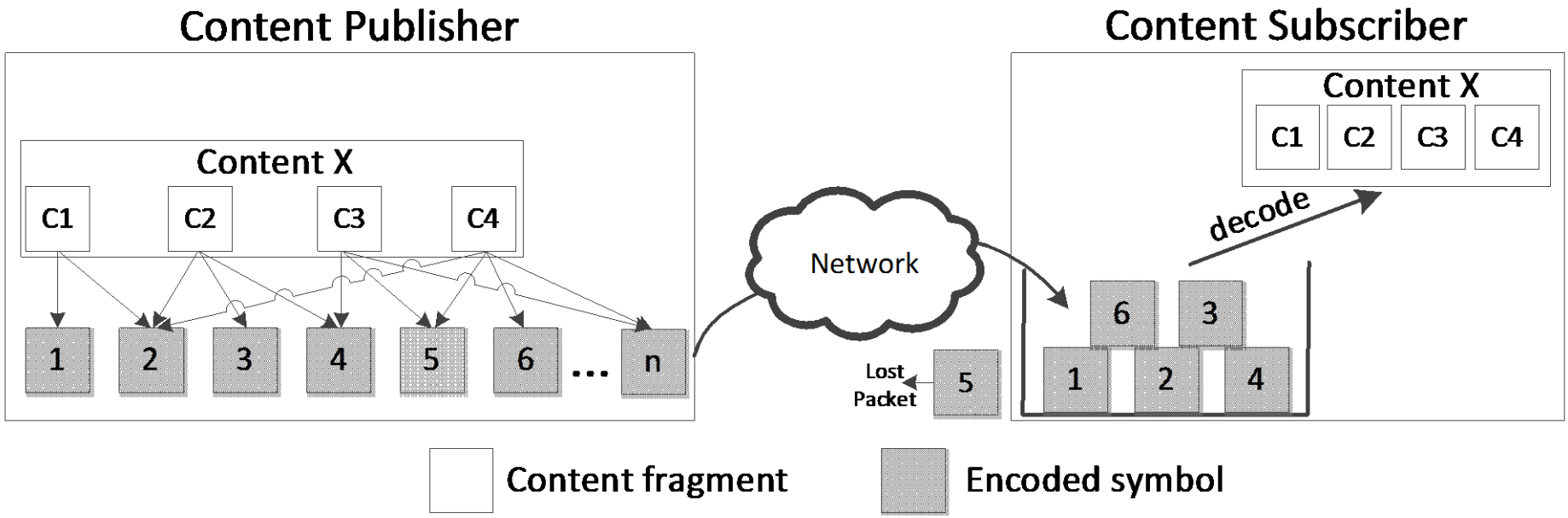
- Information-theoretic approach
- Static or dynamic content
- **Content Publisher**:
  - Initial content is fragmented (constant and known size)
  - An erasure coding can be applied (e.g. Raptor Codes) or not (e.g. Luby Codes) to the initial content
  - A very large number of encoded symbols is created as a binary combination of some fragments (XOR)
  - The art of fountain coding is to select the *degree* of each symbol!!
    - How many packets to XOR in the symbol?
    - *Neghbours' set (uniformly selected)*

Digital Fountains

- **Content Subscriber**:
  - If a symbol has *degree 1* it is decoded…
  - Using this content fragment, the degree of all symbols that contain it is decreased by 1 (XOR)

Some coding related information needs to be communicated!

Digital Fountains



- Lost packets affect only the efficiency of dissemination
- No retransmissions are required – no feedback channel
- A subscriber needs a number of symbols (slightly larger than the number of fragments) to decode the content

Key characteristics in our ICN world

- Separation of core network functions
  - Tree management is orthogonal to information dissemination (Rendezvous and Topology Management in contrast to Forwarding)
- Identification principle and Scoping
  - Embedding algorithmic relations in the identifiers (communicating coding related information in the identifier)
  - Re-use forwarding identifiers
  - All symbols are self contained --> Can be cached and replayed
- Pub/Sub model fits the notion of fountains..
  - Subscribers can join and leave the fountain – Multicasting is the norm

Basic Operation

- Based on LT codes – Domain-local dissemination – LIPSIN
- Publisher advertises /A/B
- Subscriber subscribes to /A or /A/B
- Rendezvous takes place – A TM calculates a LIPSIN ID from publisher to subscriber
- Publisher is notified that one or more subscribers exist and the digital fountain is activated

Basic Operation

- Encoded symbols are identified as */A/B/algIDn*
- The *degree* of each symbol is calculated based on a distribution (e.g. robust Soliton)
- Neighbors' set is uniformly selected based on the initial seed
- *algIDn* contains the seed and an extra bit (symbol contains last fragment)
- Re-usage of LIPSIN identifier assigned for /A/B
- The subscriber can calculate the *degree* and *neighbors'* set using the *algIDn*
- The number of fragments is learnt when a symbol with the bit *on* is received (by examining the *neighbors'* set)

Basic Operation

- Multiple subscribers can be easily supported
  - asynchronous subscriptions
  - The LIPSIN identifier for /A/B is updated (transparent to the application)
- Digital fountains can produce a very large number of unique symbols
  - with unique identifiers (the same *algIDn* defines a single symbol)
- A subscriber can join a fountain at any time
  - The content is decoded as long as the required number of symbols is received
  - The fountain stops only when no subscribers exist

Network Storage

- Network nodes can subscribe to parts of the information structure (implicitly)
- Special Link Identifiers that “point” to a caching component can be included in a LIPSIN identifier by the TM
  - Feeding the network with encoded symbols when/where needed
  - A separate control point gives power over the caching strategies
  - …instead of caching everything everywhere

Network Storage – Storage, decoding and replaying strategies

- CDN-like
  - A node decodes the content and becomes a publisher (RV is notified)
- Opportunistic
  - Nodes advertise an item without decoding it
    - When? Statistical guarantees that some content can be decoded?
    - Is it necessary? Not in a multi-publisher scenario if a source holds a decoded version…a cache stops publishing symbols when it runs out – the source will continue producing new ones
    - How to replay encoded symbols? Randomly?
- Mobility

Multi-Publisher & Multi-Path/Source

- At a given time one or more network nodes may store…
  - decoded version of a content
  - encoded symbols of a content
- Multiple LIPSIN identifiers from one or more publishers to one or more subscribers
  - Multiple sources
  - Multiple paths
  - Multiple network interfaces

Congestion Avoidance and Control

- Control plane:
  - TMs subscribe to congestion scopes
  - Forwarders publish congestion-related information
- Data plane:
  - Forwarders publish congestion notifications to end-nodes (using the respective LIPSIN identifiers)
  - Publishers publish fountain coding-related notification to subscribers

Congestion Avoidance and Control

- Control Plane: TMs can react to congestion “rebalancing” flows in the network
- Data Plane: Subscribers adjust the receiving rate (how?)
  - Asynchronous Layer Coding (RFC5775)
  - For item /A/B a number of items with IDs /A/algID(B)n are advertised
    - Specific rates for each algorithmic ID
  - A subscriber can subscribe to a number of such items
    - IDs /A/algID(B)1/algIDx, /A/algID(B)2/algIDy, /A/algID(B)3/algIDz
    - All encoded symbols for item /A/B

Thanks!