# CCN

## CCNx 1.0 – Routing in CCN
## New Solutions Needed for an Old Problem

Computer Science Laboratory
Networking & Distributed Systems

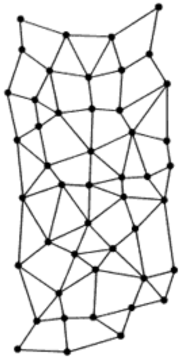
March 2014

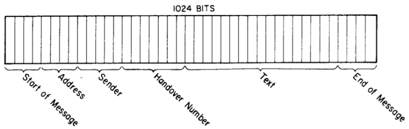
parc
A Xerox Company

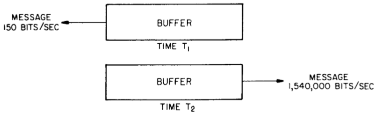
# Origins of Routing for Packet Switching

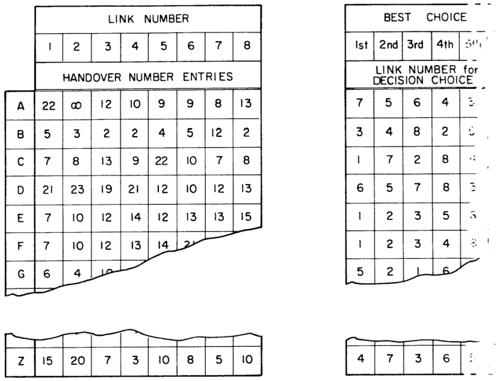
**"On Distributed Communications Series"**
Paul Baran's RAND Corporation, 1960-62.

- Packet headers, store-and-forward operation, statistical multiplexing of links
- *"Hot-potato heuristic routing doctrine"*: Primordial distance-vector routing.
- Addresses for sender and destination of packet
- Routing of packets is based on destination-based routing tables using *node addresses*.
- Destinations are routers that originate routing updates





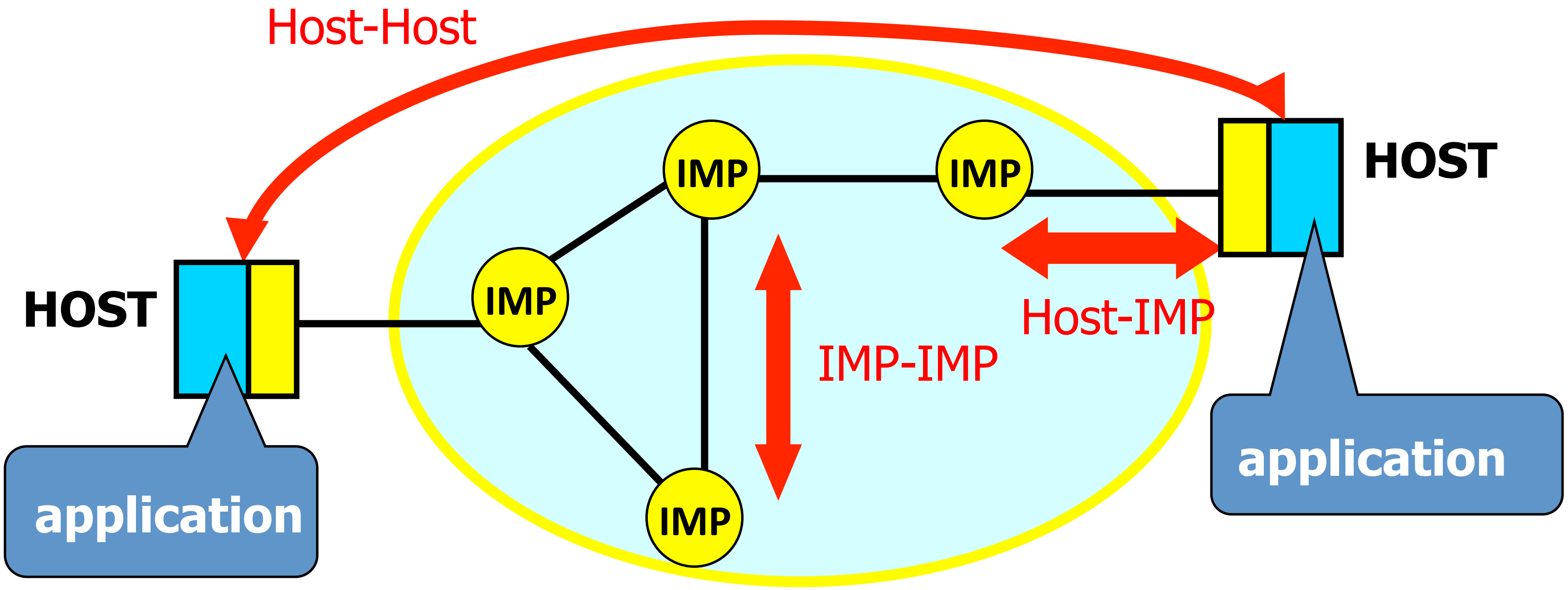




| | LINK NUMBER | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| | HANDOVER NUMBER ENTRIES | | | | | | | |
| A | 22 | ∞ | 12 | 10 | 9 | 9 | 8 | 13 |
| B | 5 | 3 | 2 | 2 | 4 | 5 | 12 | 2 |
| C | 7 | 8 | 13 | 9 | 22 | 10 | 7 | 8 |
| D | 21 | 23 | 19 | 21 | 12 | 10 | 12 | 13 |
| E | 7 | 10 | 12 | 14 | 12 | 13 | 13 | 15 |
| F | 7 | 10 | 12 | 13 | 14 | 2 | | |
| G | 6 | 4 | 10 | | | | | |
| Z | 15 | 20 | 7 | 3 | 10 | 8 | 5 | 10 |

| BEST CHOICE | | | | |
|---|---|---|---|---|
| 1st | 2nd | 3rd | 4th | 5th |
| LINK NUMBER for DECISION CHOICE | | | | |
| 7 | 5 | 6 | 4 | 3 |
| 3 | 4 | 8 | 2 | 5 |
| 1 | 7 | 2 | 8 | 4 |
| 6 | 5 | 7 | 8 | 3 |
| 1 | 2 | 3 | 5 | 6 |
| 1 | 2 | 3 | 4 | 8 |
| 5 | 2 | 1 | 6 | |
| 4 | 7 | 3 | 6 | 5 |

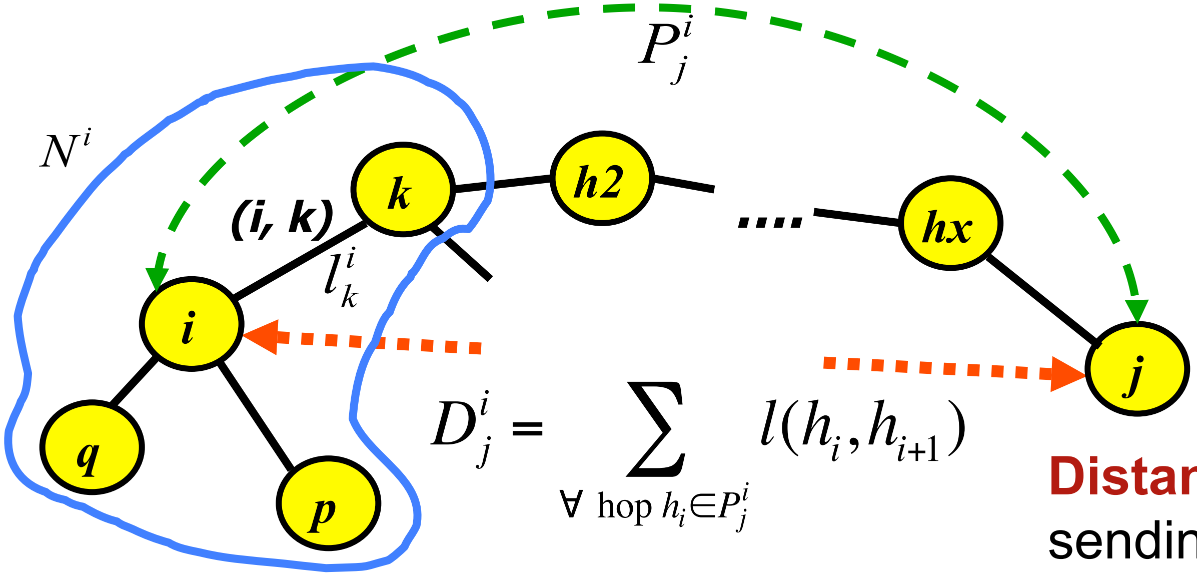
# ARPANET: Evolution from Baran's View



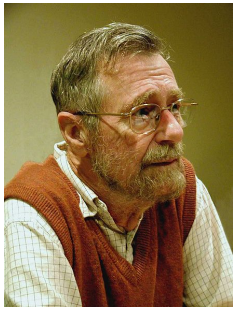
- Hosts are the destinations of the network
- Hosts and routers use addresses in *exactly* the same way
- **Same destination-based routing approach using addresses: Distance vector first, link-state approach later**
- **IMPs are the origins of updates (distances or link states)**
- **Host attached to only one IMP**

# State of The Art in Shortest-Path Routing

- ***Problem:*** Compute the path of minimum length from each router to each destination node in the network
- ***G(N, E) is the network of |N| nodes and |E| links***



$$D_j^i = \sum_{\forall \text{ hop } h_i \in P_j^i} l(h_i, h_{i+1})$$

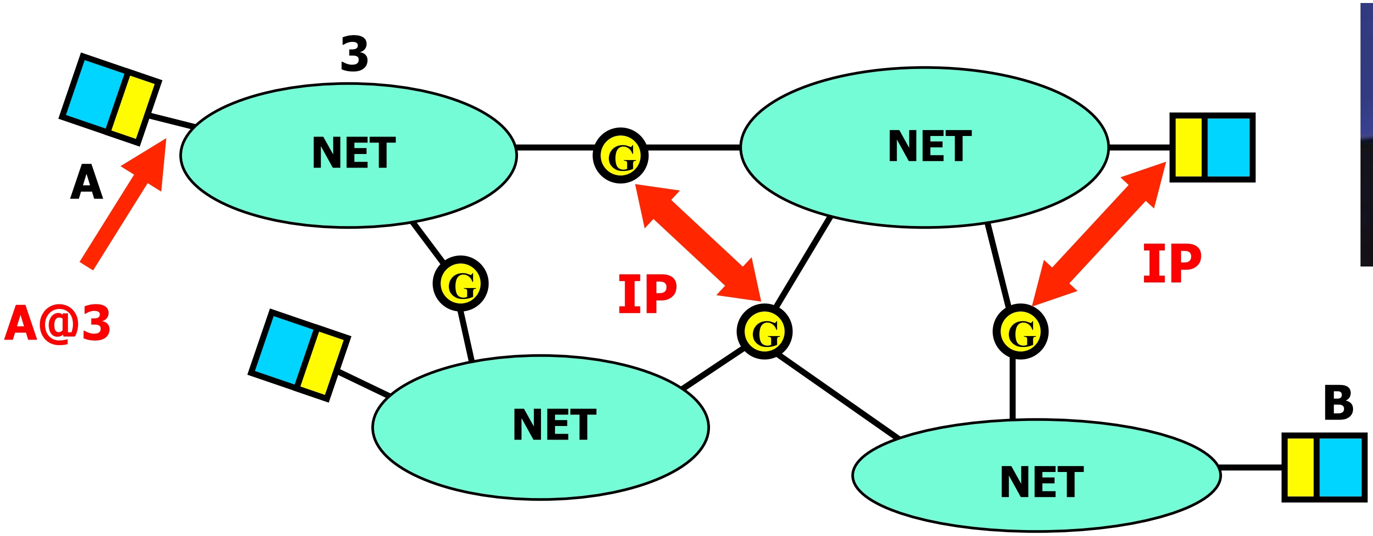


**Distances**: Node starts by sending distance to itself

**Link-state:** Node starts sending state of adjacent links

**Note that each node has a single instance in the graph**

# Internet: A Big ARPANET



- Destinations are attached to routers
- Routers "speak for" destinations [address ranges]
- Same routing algorithms are used: Distances (EIGRP, RIP), link state (OSPF), path vector (BGP)
- The choice appears obvious or trivial at first glance!
- However, the approach forces a solution in which a destination speaks for itself or has a single instance [and hence a single representative node]

# The Brave "New World"

# Information Centric Networks (ICN)

- Many papers have declared that routing in ICN is inherently different than routing in the "old" IP Internet
- Content objects are copied opportunistically in the network
- Multiple instances of the same destination.
- However, **all** existing proposals for routing in ICN use the same old routing algorithms designed for single-instance destinations!
- Is the basic routing problem in ICN really new?

# Multi-Homing and John McQuillan

- McQuillan worked on the "old" and the "new" ARPANET routing protocols:
  - McQuillan J. M., et. al., "The New Routing Algorithm for the ARPANET," *IEEE Trans. Comm.* 1980
- He and others also studied multi-homed destinations
  - J.M. McQuillan, "Enhanced Message Addressing Capabilities for Computer Networks", *Proc. IEEE,* Vol. 66, No. 11, Nov. 1978
- Same routing algorithms for single-instance destinations
- A directory is used to map identifier of "group" to identifier of each instance

# How Can We Route to Multiple Instances of The Same Thing with Today's Routing Algorithms?

Source does not know all instances

Source knows all instances

Something in between

# How Can We Route to Multiple Instances of The Same Thing with Today's Routing Algorithms?

Source does not know all instances

- Flood the network and prune as needed

# How Can We Route to Multiple Instances of The Same Thing with Today's Routing Algorithms?

## Source knows all instances

- Have each node know the topology and the location of each instance.
- Source can compute source trees to instances

# How Can We Route to Multiple Instances of The Same Thing with Today's Routing Algorithms?

Something in between (Cheat!)

- Designate a representative node between sources and destination instances
- Compute a routing tree rooted at the representative

# Sounds Familiar!

Multicasting:

- **No info:** Deering [ Flood and prune (ACM SIGCOMM 88) ]
- **All info:** McQuillan [ Link-state multicast (1978)] and Deering [ link-state multicast redone (ACM SIGCOMM 88) ]
- **Representative:** Ballardi, Francis and Crowcroft [Core Based Trees (ACM SIGCOMM 93)]; followed by Protocol Independent Multicast by Deering and many cohorts

ICN routing:

- **No info:** Intanagonwiwat et al [Directed Diffusion (Trans. Networking 2002)]
- **All info:** Jacobson [ CCN paper (CONEXT 09)]; Mahmudul-Hoque et al [NLSR (ACM ICN '13)]
- **Representative:** Carzaniga, Rosenblum, and Wolf [Content-Based Networking (Infocom 04)]

# Exciting Future!

**We do not have to assume that routing to multi-instantiated destinations has to be done the same way as for  single-instance destinations**

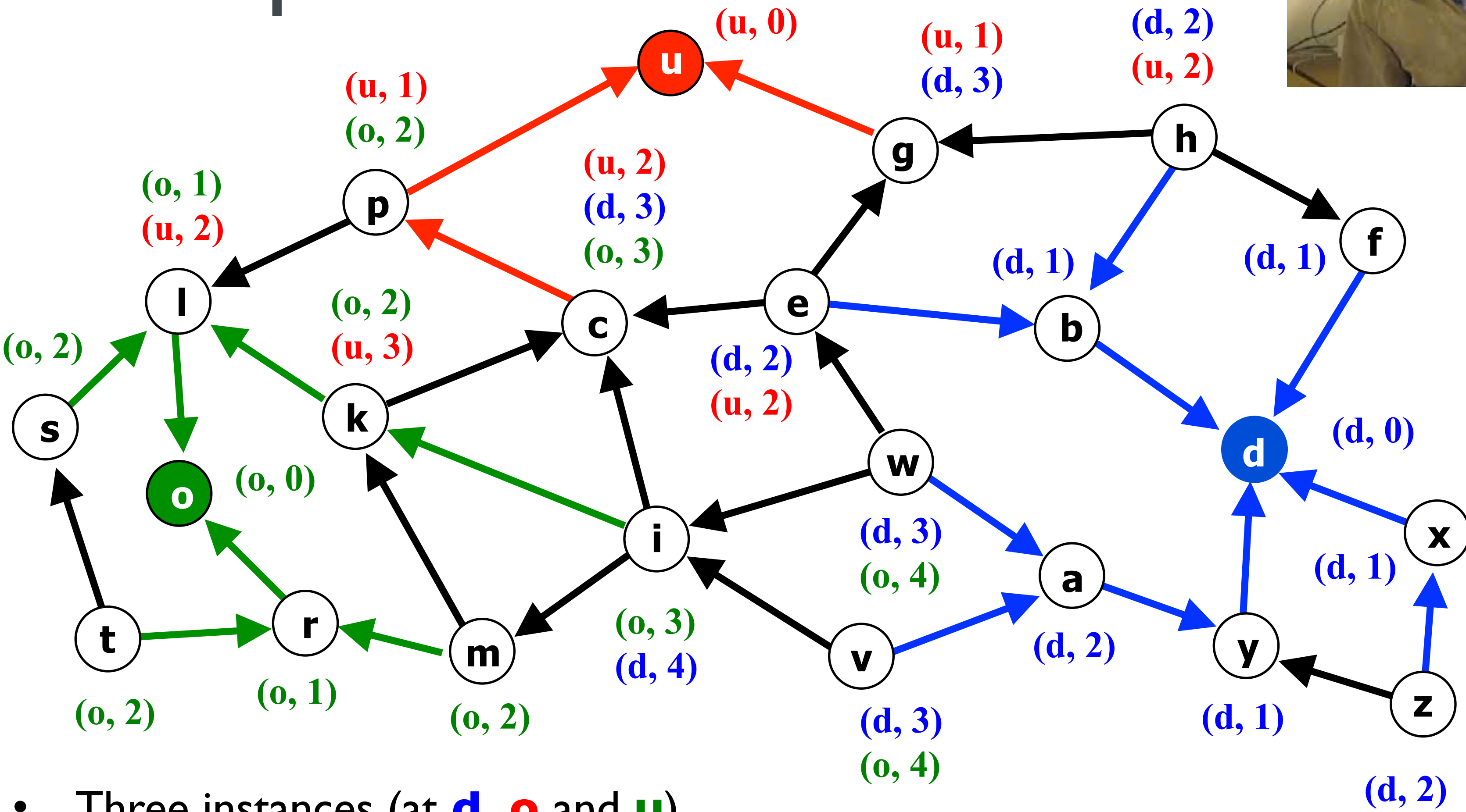
New routing algorithms are needed that:

- Route to some and all instances of a destination
- Do not flood for each instance, do not require complete information of all instances, and do not need a designated representative
- Take advantage of what we know about termination detection for single-instance destinations… It is just a distributed computation

# Basic Approach

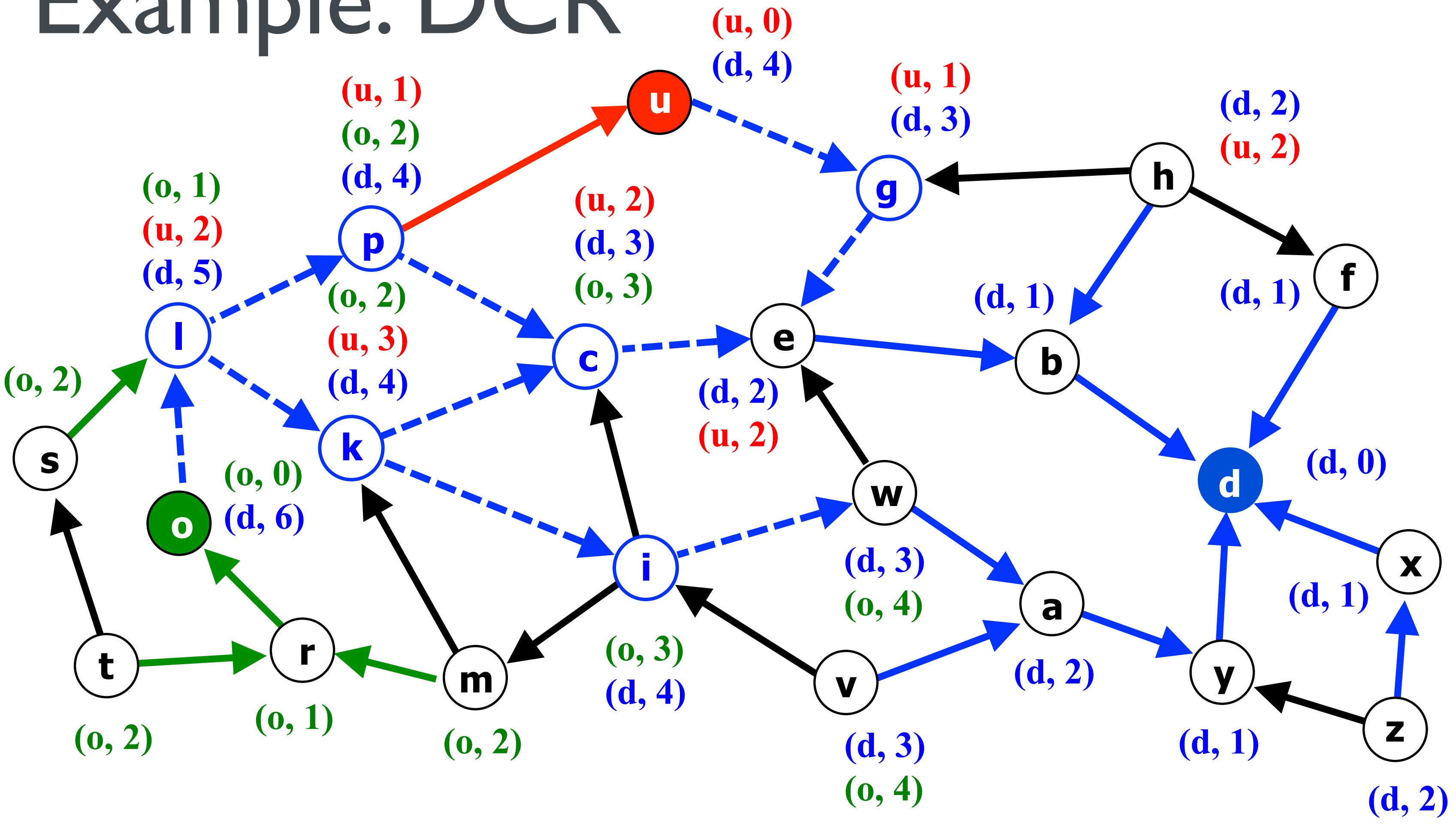
- Establish a lexicographic ordering of distances to multi-instantiated destination
- The name of a router "speaking for" a destination instance (called anchor) is an attribute used in the ordering
- Routers choose what to share with their peers (e.g.,"the best distance according to the lexicographic ordering")
- Lexicographic ordering among instances defines an instance where a DAG spanning all instances is rooted

# Example: DCR



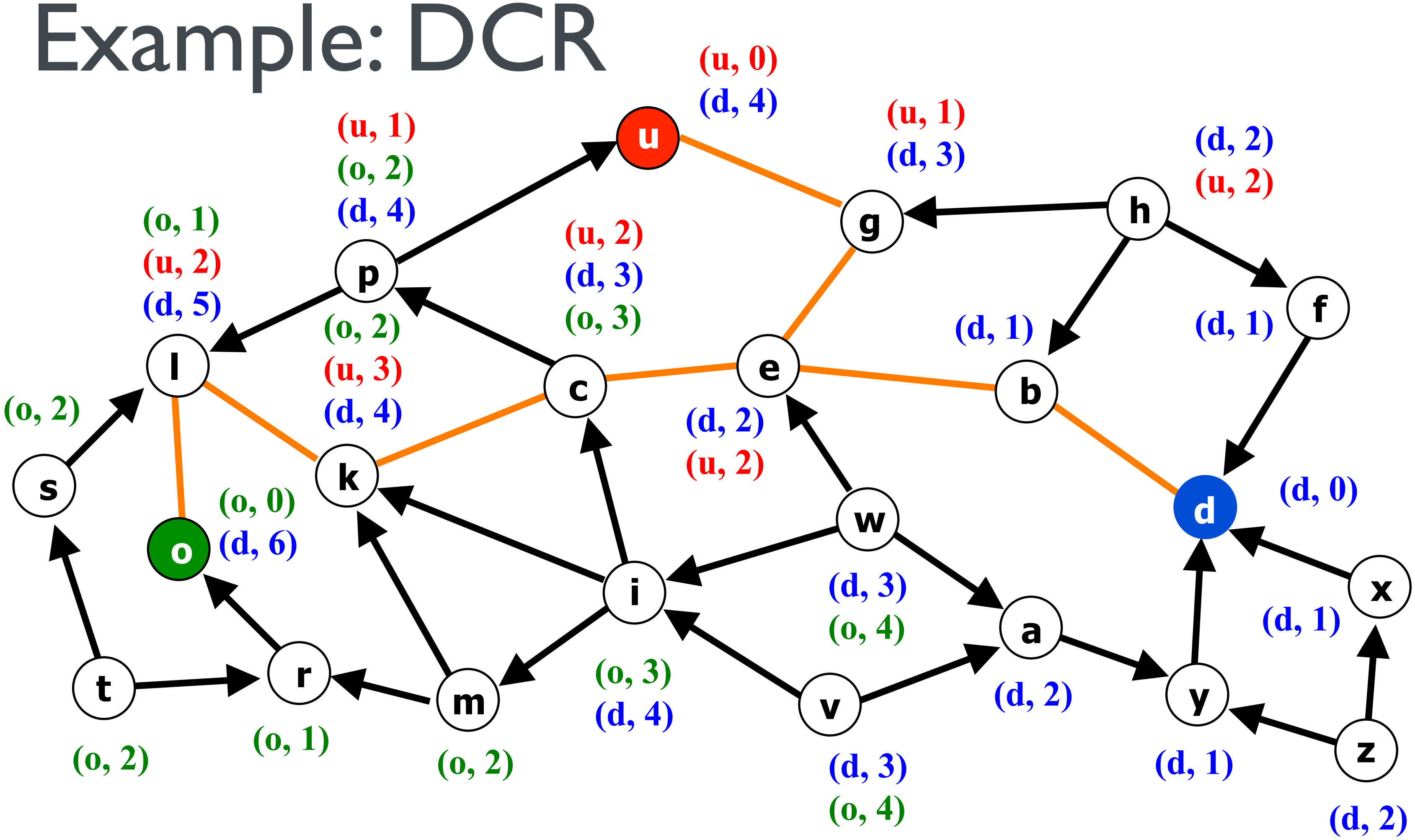
- Three instances (at **d**, **o** and **u**)
- Lexicographic ordering based on hop count to instance, ID of instance's anchor , sequence number from anchor
- Route to some [nearest] instances w/o knowing all

# Example: DCR



- Lexicographic ordering based on hop count, anchor ID, sequence number
- One instance is “best” and is hence known between instances (d < o < u)

# Example: DCR



- Routing to all instances
- We can build a routing structure spanning all instances, w/o knowing all instances, flooding, or pre-defining a “core”

# Much More Can and Should Be Done!

- Show how much better routing to multi-instantiated destinations can be compared to traditional routing approaches (NDN's NLSR, CBN)
- ns3 simulations based on the NDN simulator with thousands of routers and millions of data objects
- Play with policies and quality of information
  - You may just have access to some instances based on who you are and what you want to access
- How are autonomous systems defined in a CCN world and how is routing done?
- Secure routing: Making sure we follow valid paths to instances of a destination