Introduction to Communication Networks
The University of Kansas EECS 563
Network Routing

James P.G. Sterbenz

Department of Electrical Engineering & Computer Science
Information Technology & Telecommunications Research Center
The University of Kansas

jgps@eecs.ku.edu

http://www.ittc.ku.edu/~jgps/courses/intronets
Network Routing
Functions and Services

NR.1  Functions and services
NR.2  Routing algorithms
NR.3  PSTN routing architecture and algorithms
NR.4  Internet routing architecture and protocols
Network Layer Control
Hybrid Layer/Plane Cube

Layer 3:
routing in control plane

Layer 3: network

L8: social
L7: application
L5: session
L4: transport
L3: network
L2.5: virtual link
L2: link
L1.5: MAC
L1: physical
Network Layer
Service and Interfaces

- Network layer 3 is above link layer 2
  - addressing: network layer identifier for end systems (hosts)
  - forwarding: transfers packets hop-by-hop
    - using link layer services
    - network layer responsible for determining *which* next hop
  - routing: determination of path to forward packets
  - signalling: messages to control network layer behaviour
  - traffic management: management of traffic and congestion

- Network layer service to transport layer (L4)
  - deliver TPDUs to destination transport entity
Network Layer
Forwarding vs. Routing

- **Forwarding** transfers packets hop-by-hop
  - each switch (router) makes decision on which link to send
  - forwarding table (generally) used to make decision
  - forwarding is *per packet* decision
  - [analogy: determining which exits to take on a drive]

- **Routing** determines the path to take
  - routing algorithm independent of forwarding
  - forwarding table entries populated by routing
  - routing is (generally) not done per packet
  - [analogy: planning trip from source to destination]

*Forwarding and routing are very different*
Swatches
Functions: Routing and Signalling

- Routing
  - asynchronous w.r.t. forwarding
  - not part of critical path
  - may use topology and link state
  - uses signalling to coordinate among nodes
Network Routing
Routing Algorithms

NR.1  Functions and services
NR.2  Routing algorithms
NR.3  PSTN routing architecture and algorithms
NR.4  Internet routing architecture and protocols
Routing
Network Model

• Network is modeled as a graph $N = (V,E)$
  – nodes are vertices $v_i \in V$
  – links are edges $e_{ij} = (v_i, v_j) \in E$
    • $v_i$ and $v_j$ are neighbours
    • edge weights are costs

• Example:
  – $V = \{u, v, w, x, y, z\}$
  – $E = \{(u,v),(u,w),(u,x),(v,x),(v,w),(x,w),(x,y),(w,y),(w,z),(y,z)\}$
Routing

Network Model

- Each link (edge) has a cost $c(v_i, v_j)$
  - cost is metric of interest, e.g. hop count, latency, capacity
  - assume $c(v_i, v_j) = c(v_j, v_i)$
    - graph is not directed
  - $\forall (v_i, v_j) \notin E, c(v_j, v_i) = \infty$

- Sequence of edges (links) is path between nodes
  - $\text{path}(v_j, v_{i+n}) = (v_i, v_{i+1}, \ldots v_{j+n})$

- Cost of a path is sum of link costs
  - $c(\text{path}(v_j, v_{i+n})) = \sum c(v_j, v_{i+1})$

- Routing algorithm goal (ideal): find least cost path
  - shortest path if all link costs equal (measures hops)
Routing Algorithms
Classification by State

- State (topology and link costs)
- Global
  - complete* state located in each switch
    * this will be relaxed for hierarchical routing
  - example: link state routing
Routing Algorithms
Classification by State

- State (topology and link costs)
- Global
  - complete* state located in each switch
    * this will be relaxed for hierarchical routing
  - example: link state routing
- Decentralised
  - neighbourhood state only in each switch
  - iterative process of route computation with neighbors
  - example: distance vector routing
Routing Algorithms
Classification by Dynamicity

• Dynamicity: response to topology and traffic changes

• Static
  – routes do not change (directly)
  – changed by human operators or network management

• Dynamic (adaptive):
  – routes change in response to topology and traffic
  – route computation may occur
    • periodically
    • in direct response to changes in topology and traffic
Routing Algorithms
Classification by Load-Sensitivity

- Load-sensitivity
- Load-sensitive: routing responds to loads on links
  - load contributes to link weight
  - paths chosen along uncongested links
- Load-insensitive: routing doesn’t respond to link load
Routing Algorithms
Characteristics and Goals

• Stability
  – routes to do not *oscillate* between paths
  – harder to maintain with capacity-based dynamic routing

• Low complexity and overhead
  – processing load in switches
  – signalling message overhead on links

• Optimality
  – optimal paths for communicating nodes
  – optimal use of network resources

*Very difficult to achieve all of these!*
Static Routing
Overview

- Network engineering determines routes
  - frequently based on traffic matrix
- Forwarding tables are *provisioned* by an operator

*When does this make sense?*
Static Routing
Overview

- Network engineering determines routes
  - frequently based on traffic matrix
- Forwarding tables are *provisioned* by an operator
- Logical choice for:
  - small networks
  - networks with predictable stable traffic
Source Routing
Overview

- Node constructs entire path of packet
  - end system or edge node
  - *source route* carried in packet headers
  - routers simply pop source route stack to forward
Source Routing

Issues

• Simple in theory
  – but some entity has to create the source route
  – edge needs significant knowledge of topology

• Rarely used
  – almost never used IP option
  – some specialised problem domains
    • e.g. MANET DSR  *Lecture MW*

• Proposed for future Internet, e.g. PoMo
Dynamic Routing
Overview

• Routes change in response to topology and traffic
• Classes of dynamic routing algorithms
  – link state
  – distance vector
Link State Routing

Overview

- Use \textit{link} state to compute optimal paths
- Steps:
  - neighbor discovery
  - determine cost of each link
    - e.g. measure delay, load
  - build local link state (for distribution)
  - flood link state
  - compute routes (e.g. Dijkstra algorithm)
Link State Routing

Dijkstra Algorithm

- Network topology and link costs known to all nodes
  - accomplished via link state flooding (broadcast)
  - all nodes have consistent information
- Computes least cost paths
  - from one node (source) to all other nodes
  - gives forwarding table for that node
- Iterative
  - after $k$ iterations, know least cost path to $k$ destinations

Details and example in online notes
Link State Routing

Oscillations

- Oscillations between paths
  - link cost = load carried on link in a given direction
  - asymmetric: $c(x,y)$ not necessarily equal to $c(y,x)$

- Traffic on a given flow changes link state
  - and may cause it to switch to the other path
  - and back-and-forth again
Distance Vector Routing

Overview

• In a fully distributed manner nodes:
  – periodically send *distance vector* estimates to neighbours
  – update their own estimate when received from neighbours
  – algorithm converges and is self terminating

• Bellman-Ford equation:
  – define $d_x(y)$ as cost of least-cost path from $x$ to $y$
  – $d_x(y) = \min_v \{c(x,v) + d_v(y)\}$ over all neighbours of $x$

*Details and example in online notes

*you do not need to know the details of BF and DV*
Distance Vector Routing
Algorithm

- Iterative and asynchronous – each local iteration caused by:
  - local link cost change
  - distance vector update message from neighbour

- Distributed:
  - each node notifies neighbors only when its vector changes
  - neighbours then notify their neighbours if necessary

- Self-terminating
  - no messages when no changes need to be propagated
Distance Vector Routing

Link Cost Changes

- Link cost changes
  - node detects local link cost change
  - updates routing information
  - recalculates distance vector
  - if distance vector changes notify neighbours

- Decrease in cost converges quickly
  - “good news travels fast”

- Increase in cost converges slowly
  - “bad news travels slowly”
  - count to infinity problem
  - routing loops
Routing Algorithms

Comparison

- Link state vs. distance vector
- Comparison of characteristics

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Link State</th>
<th>Distance Vector</th>
</tr>
</thead>
<tbody>
<tr>
<td>Message complexity</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>State maintenance</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>Convergence time</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Robustness to errors</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Scalability</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Routing Algorithms Comparison

- Link state vs. distance vector
- Comparison of characteristics

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Link State</th>
<th>Distance Vector</th>
</tr>
</thead>
<tbody>
<tr>
<td>Message complexity</td>
<td>flooding link state</td>
<td>neighbour exchange</td>
</tr>
<tr>
<td>State maintenance</td>
<td>full (sub-)network</td>
<td>neighbours</td>
</tr>
<tr>
<td>Convergence time</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>Robustness to errors</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Scalability</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Routing Algorithms

Comparison

- Link state vs. distance vector
- Comparison of characteristics

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Link State</th>
<th>Distance Vector</th>
</tr>
</thead>
<tbody>
<tr>
<td>Message complexity</td>
<td>flooding link state</td>
<td>neighbour exchange</td>
</tr>
<tr>
<td>State maintenance</td>
<td>full (sub-)network</td>
<td>neighbours</td>
</tr>
<tr>
<td>Convergence time</td>
<td>fast</td>
<td>slow</td>
</tr>
<tr>
<td>Robustness to errors</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>Scalability</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Routing Algorithms

### Comparison

- Link state vs. distance vector
- Comparison of characteristics

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Link State</th>
<th>Distance Vector</th>
</tr>
</thead>
<tbody>
<tr>
<td>Message complexity</td>
<td>flooding link state</td>
<td>neighbour exchange</td>
</tr>
<tr>
<td>State maintenance</td>
<td>full (sub-)network</td>
<td>neighbours</td>
</tr>
<tr>
<td>Convergence time</td>
<td>fast</td>
<td>slow</td>
</tr>
<tr>
<td>Robustness to errors</td>
<td>errors limited to link(s)</td>
<td>errors propagate</td>
</tr>
<tr>
<td>Scalability</td>
<td>?</td>
<td>?</td>
</tr>
</tbody>
</table>
Routing Algorithms
Comparison

- Link state vs. distance vector
- Comparison of characteristics

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Link State</th>
<th>Distance Vector</th>
</tr>
</thead>
<tbody>
<tr>
<td>Message complexity</td>
<td>flooding link state</td>
<td>neighbour exchange</td>
</tr>
<tr>
<td>State maintenance</td>
<td>full (sub-)network</td>
<td>neighbours</td>
</tr>
<tr>
<td>Convergence time</td>
<td>fast</td>
<td>slow</td>
</tr>
<tr>
<td>Robustness to errors</td>
<td>errors limited to link(s)</td>
<td>errors propagate</td>
</tr>
<tr>
<td>Scalability</td>
<td>reasonable (with hierarchy)</td>
<td>poor (long convergence)</td>
</tr>
</tbody>
</table>
Routing Algorithms

Scalability Challenges

• Scalability of routing algorithms
  – processing and bandwidth: message complexity
  – memory: state maintained

• Link state scalability challenges
  – flooding of link state in large network
  – large topology data bases

• Distance vector scalability challenges
  – long path lengths

Solution?
Routing Algorithms

Hierarchical Routing Concepts

- Hierarchy
  - divide network into clusters
  - isolate full topology and link state in lowest layers
  - higher layers aggregate
Network Routing

PSTN Routing Architecture and Algorithms

NR.1 Functions and services
NR.2 Routing algorithms
NR.3 PSTN routing architecture and algorithms
NR.4 Internet routing architecture and protocols
PSTN Routing
Background and Overview

- PSTN evolved from a strict hierarchy
  - hierarchical E.164 addressing
  - with a (almost) single entity in technical control in the US
- Initial routing algorithm matches hierarchy
  - HIER
- Later algorithms relaxed strict hierarchy
  - better load balancing
  - necessary for NPA overlay and number portability
US PSTN
Traditional Hierarchical Network Structure

final trunk group
(to class 4 long distance switching)

tandem trunk group
tandem office

local trunk group
direct trunk group

local (class 5) switches
direct trunk group

tandem office
local office

local loops

PSTN Fixed Hierarchical Routing

Overview

• Fixed Hierarchical routing (HIER)
  – static routing algorithm based on strict hierarchy

• Switch class hierarchy
  5: local
  4: toll
  3: primary
  2: regional
  1: sectional

• Link types
  – final
  – high usage
PSTN Hierarchical Routing

Link Types

- **Final links**
  - each switch *homed* to single switch above
  - top level interconnected
  - *ladder*: vertical final link set
PSTN Hierarchical Routing

Link Types

- **High-usage links**
  - additional direct trunks
  - deployed as needed based on traffic matrix
  - reduce num. of hops below max. of 9
  - reduce bottlenecks in lower classes
PSTN Fixed Hierarchical Routing

Routing Design Rules

• Routing *design rules* define routing algorithm
  – constrain space of alternate choice and prevent looping

• Rules
  – two-ladder limit
  – intraladder direction
  – multiple switching function
  – one-level limit
  – switch low
  – directional routing
  – single route
  – alternate-route selection
PSTN Fixed Hierarchical Routing

Interladder Route Selection

- Order of link selection
  - assuming trunk exists
  - capacity available
- Goals
  - minimise number of hops
  - distribute load in lower classes

based on [Girad-1990 Fig 2.24]
PSTN Fixed Hierarchical Routing

Interladder Route Selection

• Order of link selection
  – assuming trunk exists
  – capacity available

• Hunt sequence
  – direct high-usage link
  • if switch LAMA capable
    (local automatic message accounting)
PSTN Fixed Hierarchical Routing
Interladder Route Selection

- **Order of link selection**
  - assuming trunk exists
  - capacity available

- **Hunt sequence**
  - direct high-usage link
  - high-usage class 4 links
PSTN Fixed Hierarchical Routing

Interladder Route Selection

- Order of link selection
  - assuming trunk exists
  - capacity available

- Hunt sequence
  - direct high-usage link
  - high-usage class 4 links
PSTN Fixed Hierarchical Routing
Interladder Route Selection

- Order of link selection
  - assuming trunk exists
  - capacity available
- Hunt sequence
  - direct high-usage link
  - high-usage class 4 links
PSTN Fixed Hierarchical Routing

Interladder Route Selection

- Order of link selection
  - assuming trunk exists
  - capacity available
- Hunt sequence
  - direct high-usage link
  - high-usage class 4 links
  - high-usage class 3 links
PSTN Fixed Hierarchical Routing

Interladder Route Selection

• Order of link selection
  – assuming trunk exists
  – capacity available

• Hunt sequence
  – direct high-usage link
  – high-usage class 4 links
  – high-usage class 3 links
PSTN Fixed Hierarchical Routing

Interladder Route Selection

- Order of link selection
  - assuming trunk exists
  - capacity available

- Hunt sequence
  - direct high-usage link
  - high-usage class 4 links
  - high-usage class 3 links
PSTN Fixed Hierarchical Routing
Interladder Route Selection

- **Order of link selection**
  - assuming trunk exists
  - capacity available

- **Hunt sequence**
  - direct high-usage link
  - high-usage class 4 links
  - high-usage class 3 links
PSTN Fixed Hierarchical Routing

Interladder Route Selection

- Order of link selection
  - assuming trunk exists
  - capacity available

- Hunt sequence
  - direct high-usage link
  - high-usage class 4 links
  - high-usage class 3 links
PSTN Fixed Hierarchical Routing

Interladder Route Selection

• Order of link selection
  – assuming trunk exists
  – capacity available

• Hunt sequence
  – direct high-usage link
  – high-usage class 4 links
  – high-usage class 3 links
  – high-usage class 2 links
PSTN Fixed Hierarchical Routing

Interladder Route Selection

- Order of link selection
  - assuming trunk exists
  - capacity available

- Hunt sequence
  - direct high-usage link
  - high-usage class 4 links
  - high-usage class 3 links
  - high-usage class 2 links
PSTN Fixed Hierarchical Routing

Interladder Route Selection

- Order of link selection
  - assuming trunk exists
  - capacity available
- Hunt sequence
  - direct high-usage link
  - high-usage class 4 links
  - high-usage class 3 links
  - high-usage class 2 links
PSTN Fixed Hierarchical Routing

Interladder Route Selection

- Order of link selection
  - assuming trunk exists
  - capacity available

- Hunt sequence
  - direct high-usage link
  - high-usage class 4 links
  - high-usage class 3 links
  - high-usage class 2 links
PSTN Fixed Hierarchical Routing

Interladder Route Selection

- Order of link selection
  - assuming trunk exists
  - capacity available
- Hunt sequence
  - direct high-usage link
  - high-usage class 4 links
  - high-usage class 3 links
  - high-usage class 2 links
PSTN Fixed Hierarchical Routing

Interladder Route Selection

• Order of link selection
  – assuming trunk exists
  – capacity available

• Hunt sequence
  – direct high-usage link
  – high-usage class 4 links
  – high-usage class 3 links
  – high-usage class 2 links
PSTN Fixed Hierarchical Routing

Interladder Route Selection

• Order of link selection
  – assuming trunk exists
  – capacity available

• Hunt sequence
  – direct high-usage link
  – high-usage class 4 links
  – high-usage class 3 links
  – high-usage class 2 links
PSTN Fixed Hierarchical Routing

Interladder Route Selection

- Order of link selection
  - assuming trunk exists
  - capacity available

- Hunt sequence
  - direct high-usage link
  - high-usage class 4 links
  - high-usage class 3 links
  - high-usage class 2 links
  - high-usage class 1 links
PSTN Fixed Hierarchical Routing

Interladder Route Selection

- Order of link selection
  - assuming trunk exists
  - capacity available
- Hunt sequence
  - direct high-usage link
  - high-usage class 4 links
  - high-usage class 3 links
  - high-usage class 2 links
  - high-usage class 1 links
PSTN Fixed Hierarchical Routing
Interladder Route Selection

- Order of link selection
  - assuming trunk exists
  - capacity available

- Hunt sequence
  - direct high-usage link
  - high-usage class 4 links
  - high-usage class 3 links
  - high-usage class 2 links
  - high-usage class 1 links
PSTN Fixed Hierarchical Routing

Interladder Route Selection

- Order of link selection
  - assuming trunk exists
  - capacity available
- Hunt sequence
  - direct high-usage link
  - high-usage class 4 links
  - high-usage class 3 links
  - high-usage class 2 links
  - high-usage class 1 links
PSTN Fixed Hierarchical Routing

Interladder Route Selection

• Order of link selection
  – assuming trunk exists
  – capacity available

• Hunt sequence
  – direct high-usage link
  – high-usage class 4 links
  – high-usage class 3 links
  – high-usage class 2 links
  – high-usage class 1 links
PSTN Fixed Hierarchical Routing
Interladder Route Selection

- Order of link selection
  - assuming trunk exists
  - capacity available

- Hunt sequence
  - direct high-usage link
  - high-usage class 4 links
  - high-usage class 3 links
  - high-usage class 2 links
  - high-usage class 1 links
PSTN Fixed Hierarchical Routing

Interladder Route Selection

- Order of link selection
  - assuming trunk exists
  - capacity available
- Hunt sequence
  - direct high-usage link
  - high-usage class 4 links
  - high-usage class 3 links
  - high-usage class 2 links
  - high-usage class 1 links
PSTN Fixed Hierarchical Routing

Interladder Route Selection

- Order of link selection
  - assuming trunk exists
  - capacity available

- Hunt sequence
  - direct high-usage link
  - high-usage class 4 links
  - high-usage class 3 links
  - high-usage class 2 links
  - high-usage class 1 links
PSTN Fixed Hierarchical Routing

Interladder Route Selection

- Order of link selection
  - assuming trunk exists
  - capacity available

- Hunt sequence
  - direct high-usage link
  - high-usage class 4 links
  - high-usage class 3 links
  - high-usage class 2 links
  - high-usage class 1 links
  - final links (9 hops)
    - final choice possible

Diagram:

1 -> 25 -> 1
1 -> 2
2 -> 3
3 -> 4
4 -> 5
5
PSTN Fixed Hierarchical Routing

Intraladder Route Selection

- **Order of link selection**
  - assuming trunk exists
  - capacity available
  - given intra-link selection

- **Goals**
  - minimise number of hops
  - reduce final link load
PSTN Fixed Hierarchical Routing

Intraladder Route Selection

• Order of link selection
  – assuming trunk exists
  – capacity available
  – given intra-link selection

• Hunt sequence
  – direct high-usage trunk
PSTN Fixed Hierarchical Routing

Intraladder Route Selection

- Order of link selection
  - assuming trunk exists
  - capacity available
  - given intra-link selection

- Hunt sequence
  - direct high-usage trunk
  - reduce class 1 hops
PSTN Fixed Hierarchical Routing

Intraladder Route Selection

- Order of link selection
  - assuming trunk exists
  - capacity available
  - given intra-link selection

- Hunt sequence
  - direct high-usage trunk
  - reduce class 1 hops
PSTN Fixed Hierarchical Routing

Intraladder Route Selection

- Order of link selection
  - assuming trunk exists
  - capacity available
  - given intra-link selection

- Hunt sequence
  - direct high-usage trunk
  - reduce class 1 hops
  - reduce total/class 2 hops
PSTN Fixed Hierarchical Routing

Intraladder Route Selection

- Order of link selection
  - assuming trunk exists
  - capacity available
  - given intra-link selection

- Hunt sequence
  - direct high-usage trunk
  - reduce class 1 hops
  - reduce class 2 hops
PSTN Fixed Hierarchical Routing

Intraladder Route Selection

- Order of link selection
  - assuming trunk exists
  - capacity available
  - given intra-link selection

- Hunt sequence
  - direct high-usage trunk
  - reduce class 1 hops
  - reduce class 2 hops
PSTN Fixed Hierarchical Routing

Intraladder Route Selection

- Order of link selection
  - assuming trunk exists
  - capacity available
  - given intra-link selection

- Hunt sequence
  - direct high-usage trunk
  - reduce class 1 hops
  - reduce class 2 hops
PSTN Fixed Hierarchical Routing

Summary

Advantages?
PSTN Fixed Hierarchical Routing

Summary

- Advantages
  - simple: directly related to network topology
  - easy to implement in early relay-logic switches

Disadvantages?
PSTN Fixed Hierarchical Routing

Summary

- **Advantages**
  - simple: directly related to network topology
  - easy to implement in early relay-logic switches

- **Disadvantages**
  - inflexible: class 5 switches cannot serve as transit nodes
  - unable to adapt to time-varying load

*Alternatives?*
PSTN Routing
Non-Hierarchical Topology

- Modern PSTN lacks strict hierarchy of Bell System
  - divestiture and deregulation
    - multiple IXCs connecting ILECs and CLECs
  - NPA overlays
  - LNP (local number portability)
PSTN Routing
Non-Hierarchical and Dynamic Routing

- **Non-Hierarchical routing**
  - removes requirement routing matches physical hierarchy
  - note: different from hierarchy for scalability (e.g. PNNI)
  - planned before divestiture to increase efficiency of network

- **Dynamic routing**
  - permit routing to adapt to traffic
PSTN Non-Hierarchical Routing

DNHR Overview

- DNHR (dynamic non-hierarchical routing)
- AT&T long distance network
  - Bell Labs research in early 1980s
  - deployed in AT&T long distance network 1984–1991
  - former classes 3 – 1 switches
- Network is a mesh of switches
- Each switch pair connected by:
  - direct primary path
  - alternate two-hop paths
PSTN Real-Time Network Routing

RTNR Overview

- RTNR (real-time network routing)
  - developed by Bell Labs in late 1980s to improve on DNHR

- Motivation
  - DNHR was not adaptive enough
  - off-line computations and administration costly
  - increased switch processing and memory capabilities

- Forwarding tables
  - change dynamically based on traffic conditions

- Routing
  - fully distributed algorithm
  - paths computed per call
Network Routing

Internet Routing

NR.1  Functions and services
NR.2  Routing algorithms
NR.3  PSTN routing architecture and algorithms
NR.4  Internet routing architecture and protocols
Internet Routing Topology
Current Internet Structure

tier-2 providers

IXP

peering points

IXP

local ISPs

access lines

tier-1 providers
Internet Routing Topology
AS Intra/Inter-Domain Hierarchy

- *Autonomous systems* (ASs) or *domains*
  - administrative autonomy
  - improves scalability

- Two-layer major hierarchy
  - intradomain within AS
  - interdomain between ASs

- Routing protocols loads forwarding tables
Internet Routing

Intradomain Routing and Forwarding

• Intra-AS or *intradomain* routing
  – IGP (interior gateway protocol)

• IGP determines intra-AS entries
  – determines paths within AS
  – loads forwarding tables within AS
  – IP forwards

• Each AS can choose the IGP of its choice
  – may be hierarchical *within* AS

• Example IGPs
  – RIP, OSPF, ISIS, IGRP/EIGRP
Internet Routing
Interdomain Routing and Forwarding

• Inter-AS or *interdomain* routing
  – EGP (exterior gateway protocol)
• EGP determines inter-AS entries in edge nodes
  – determine inter-AS reachability for particular AS IP blocks
  – loads edge router forwarding tables
  – must also propagate within AS and load forwarding tables
• All Internet ASs *must* agree on the EGP
  – BGP-4: border gateway protocol is *de facto* standard
  – evolution difficult; replacement probably impossible
Intradomain Routing

Intradomain Protocols

- **RIP**: routing information protocol
  - original IGP; no longer in wide use
- **EIGRP**: (enhanced) interior gateway routing protocol
  - Cisco proprietary replacement for RIP
- **ISIS**: intermediate system – intermediate system
  - OSI IGP
- **OSPF**: open shortest path first
  - IETF replacement for RIP
- **OSPF** or **ISIS** used by virtually all service providers
  - distance vector convergence too slow
Intradomain Routing

ISIS Overview

- ISIS: intermediate system to intermediate system
  - used by many service providers
  - was available before OSPF
- OSI routing protocol
  - derived from DECnet phase V routing
  - adopted by ISO for CLNP (connectionless networking)
- Link state routing protocol
  - messages transported directly over IP
    - ISIS protocol ID = 124
  - very similar to OSPF in operation
Intradomain Routing

OSPF Link State

- **OSPF** is a link state routing protocol [RFC 2328/5340, v6]
  - IETF open development process
  - messages transported directly over IP: protocol ID = 89
- Dijkstra least-cost path algorithm
  - *shortest path first* (SPF)
  - link costs
    - configured by network administrator
    - set to 1 for minimum hop routing
Intradomain Routing
OSPF Initialisation and Operation

• Initialisation
  – hello establishes link adjacencies to all neighbours
    – and maintains liveness
  – database description loads topology information

• Operation
  – link state advertisements *flooded* to AS
    • directed link weights use multiple parameters
    • link costs configured by network administrator
    • 1 for minimum hop routing
  – Dijkstra least-cost path algorithm
Intradomain Routing
OSPF Link State Advertisements

- LSAs (link state advertisements)
  - reliable flooding within AS
    - ACKs returned
- LSAs flooded
  - whenever link state changes
  - periodically (at least 30 min. interval)
Intradomain Routing

OSPF LSA Header

- LSA header (after common header)
  - LS age: time since LSA originated [s]
  - options
  - type
  - link state ID
  - advertising router
  - LS sequence number
  - LS checksum
  - length

todo: packet format figure
Intradomain Routing

OSPF Features

• Security:
  – all OSPF messages authenticated

• Multipath routing
  – multiple equal-cost paths supported
  – can cause problems to end-to-end protocols
    • effect on TCP packet ordering

• Hierarchy
  – 2-level scales to larger ASs

• Multicast routing
  – MOSPF: multicast OSPF [RFC 1584]
Intradomain Routing

OSPF Hierarchy

- OSPF supports 2-layer hierarchy
- AS divided into *areas* connected by the *backbone*
  - backbone area ID 0.0.0.0
- LSAs flooded only within area
  - or within backbone
- Border routers summarise costs within area
  - advertise to other border routers
- Boundary routers connect to other ASs
  - results in 3-layer Internet hierarchy:
    BGP / OSPF backbone / OSPF area

todo: figure
AS Routing Hierarchy
Motivation for Different Protocols

• Motivation for different inter- and intra-AS protocols
• Flexibility
  – AS boundaries are administrative reality
  – allows each AS to choose its own IGP
  – but doesn’t allow integrated hierarchy with IGPs
• Policy
  – inter-AS peering and transit a business decision
  – BGP policy not well designed from beginning
    • policies may never converge
    • convergence dependent on message order and link state
    • not possible to compute if policies will converge
Interdomain Routing

BGP Overview

- Border gateway protocol (BGP) [RFC 2328 / STD 0054]
- IETF interdomain routing protocol (EGP)
  - de facto standard; all ASs must use to interconnect
  - current version is BGP4
  - BGP replaced initial EGP [RFC 0904]
- BGP has many flaws
  - many papers on how BGP is broken and proposals to fix
  - but BGP is part of the (interdomain) Internet hourglass
    - as hard to evolve or replace as IP
Interdomain Routing
Current Internet Structure

tier-2 providers

IXP

IXP

peering points

tier-1 providers

local ISPs

access lines
Interdomain Routing
BGP Path Vector

- *Path vector* routing protocol
- Similar to distance vector
  - paths advertised instead of distance metrics
  - sequence of AS numbers
    - assigned by RIPE, ARIN, etc.
- Propagates *reachability* information
  - ASs advertise their existence to Internet
  - policy determines which choices an AS makes
- Messages transported over TCP
Interdomain Routing

BGP Sessions

- **BGP session** is association between *peers*
  - **eBGP**: exterior BGP session between ASs
  - **iBGP**: interior BGP session within AS
    - fully connected mesh
    - multihop TCP if direct link doesn’t exist
    - allows route advertisements to reach every node

[Kurose–Ross p.380]
Interdomain Routing
BGP Route Advertisements

- Route *advertisements*
  - promise to forward to destination
  - IP prefixes are aggregated
Interdomain Routing
BGP Reachability Example

- BGP advertisement example
  - propagation of AS3:3A reachability into AS2
Interdomain Routing

BGP Reachability Example

- BGP advertisement example
  - propagation of AS3:3A reachability into AS2
- AS3 prefix reachability to AS1 using eBGP 3A→1C
  - advertises
Interdomain Routing
BGP Reachability Example

• BGP advertisement example
  – propagation of AS3:3A reachability into AS2
• AS3 prefix reachability to AS1 using eBGP 3A→1C
• 1C distributes new prefix route using iBGP in AS1
Interdomain Routing

BGP Reachability Example

- BGP advertisement example
  - propagation of AS3:3A reachability into AS2
- AS3 prefix reachability to AS1 using eBGP 3A→1C
- 1C distributes new prefix route using iBGP in AS1
- 1B re-advertise new prefix to AS2 using eBGP 1B→2A
Interdomain Routing  
BGP Reachability Example

- BGP advertisement example  
  - propagation of AS3:3A reachability into AS2  
- AS3 prefix reachability to AS1 using eBGP 3A→1C  
- 1C distributes new prefix route using iBGP in AS1  
- 1B re-advertise new prefix to AS2 using eBGP 1B→2A  
- New prefix received creates fwd table entry to AS3
Intradomain Routing
 Routes and Path Attributes

• BGP prefix advertisements include BGP attributes
  – prefix + attributes = route

• Important attributes:
  – AS-PATH
    • AS numbers through which advertisement passed
  – NEXT-HOP
    • specific internal-AS router to next-hop AS
    • may be multiple links from current AS to next-hop-AS

• When edge router receives route advertisement
  – use import policy to accept/decline
Intradomain Routing
BGP Route Selection

- Router may learn about multiple routes to a prefix
- Elimination rules:
  - local preference value attribute
    - policy decision
  - shortest AS-PATH
  - closest NEXT-HOP router
    - hot potato routing
  - additional policy criteria
Intradomain Routing

BGP Messages

- BGP messages exchanged over TCP
  - OPEN
    - opens TCP connection to peer & authenticates sender
  - UPDATE
    - advertises new path (or withdraws old)
  - KEEPALIVE
    - keeps connection alive in absence of UPDATES
    - also acknowledges OPEN request
  - NOTIFICATION
    - reports errors in previous message
    - also used to close connection

todo: packet format figure
Intradomain Routing

BGP Example

- **Topology**
  - provider networks AS 1,2,3
    - e.g. tier-1 or tier-2
  - subscriber networks AS 70,80,90
    - e.g. tier \( n-1 \), enterprise, campus
  - AS80 dual-homed
    *why?*

[Kurose–Ross p.382]
Intradomain Routing
BGP Example: Multihoming

• Topology
  – provider networks AS 1,2,3
    • e.g. tier-1 or tier-2
  – subscriber stub networks AS 70,80,90
    • e.g. tier $n-1$, enterprise, campus
  – AS80 dual-homed
    • for resilience if one service provider fails or link goes down

[Kurose–Ross p.382]
Intradomain Routing
BGP Policy Example: Transit Traffic

- Subscriber network policy: transit traffic

*Is AS80 likely to want to route between AS2 and AS3?*

*Why?*

[Kurose–Ross p.382]
Intradomain Routing

BGP Policy Example: Transit Traffic

- Subscriber network policy: transit traffic
  - AS80 is unlikely to route transit traffic between AS2 and AS3
    - stub network only sources or sinks traffic
    - tier-$n-1$ not engineered to transit tier-$n$ traffic
    - no economic benefit

*How is this accomplished?*

[Kurose–Ross p.382]
Intradomain Routing
BGP Policy Example: Transit Traffic

- Subscriber network policy: transit traffic
  - AS80 is unlikely to route transit traffic between AS2 and AS3
- AS80 route advertisements
  - will not advertise $AS80 \rightarrow AS3 \rightarrow AS90$ to AS2

[Kurose–Ross p.382]
• Provider policy examples
  – AS1 advertises 1→70 to AS2
  – AS2 advertises 2→1→70 to AS80

*should AS2 advertise 2→1→70 to AS3?*
Intradomain Routing

BGP Policy Example: Provider Policy

• Provider policy examples
  – AS1 advertises $1\rightarrow 70$ to AS2
  – AS2 advertises $2\rightarrow 1\rightarrow 70$ to AS80
  – AS2 not likely to advertise $2\rightarrow 1\rightarrow 70$ to AS3
    • neither AS70 nor AS3 are AS2’s customers
    • AS2 wants to force AS3 to route to AS70 via AS1
    • AS2 wants to route only to and from its own customers

[Kurose–Ross p.382]
## Network Routing

### Key Acronyms

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HIER</td>
<td>PSTN hierarchical routing</td>
</tr>
<tr>
<td>DNHR</td>
<td>PSTN dynamic non-hierarchical routing</td>
</tr>
<tr>
<td>AS</td>
<td>autonomous system</td>
</tr>
<tr>
<td>OSPF</td>
<td>open shortest-path routing</td>
</tr>
<tr>
<td>ISIS</td>
<td>intermediate-system to intermediate-system</td>
</tr>
<tr>
<td>LSA</td>
<td>link state advertisement</td>
</tr>
<tr>
<td>BGP</td>
<td>border gateway protocol</td>
</tr>
</tbody>
</table>
Communication Networks

Acknowledgements

Some material in these foils comes from the textbook supplementary materials:

- Kurose & Ross,
  *Computer Networking: A Top-Down Approach Featuring the Internet*

- Sterbenz & Touch,
  *High-Speed Networking: A Systematic Approach to High-Bandwidth Low-Latency Communication*
  
  http://hsn-book.sterbenz.org