iBGP Route Reflectors topologies

Eduardo Grampín
Universidad Carlos III de Madrid

Outline

- Practical design guidelines
- Correct and scalable proposals
- Recent IETF proposals
- Others

Practical design guidelines

- Bates [RFC 4456]
- Zhang [BGPDesign]

Bates

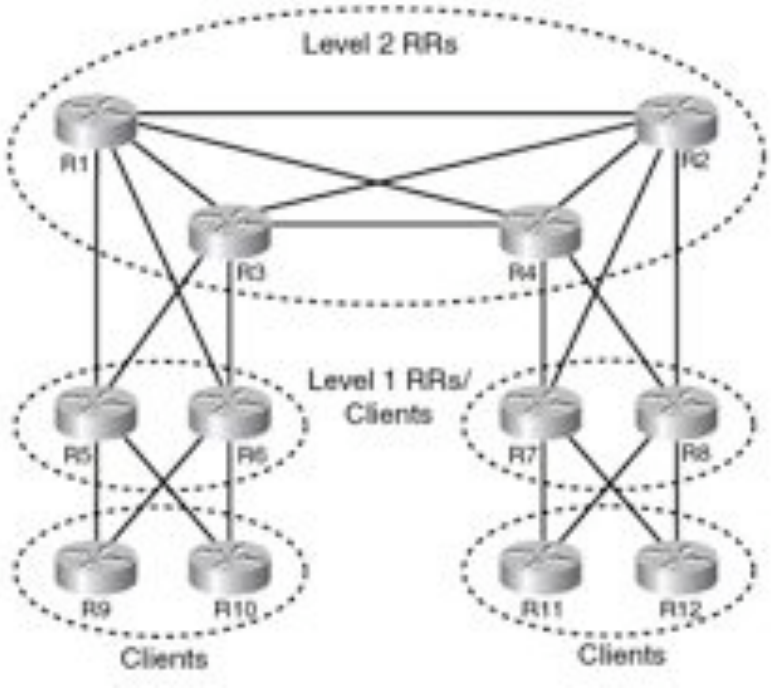
- "Bates1"
  - The most connected router in each PoP is selected to be the RR
  - Each router is a client of the RR in its PoP
  - A full-mesh of iBGP sessions is established between the RRs
  - Finally, there is a full-mesh of iBGP sessions between all the routers in a PoP
- "Bates2"
  - The two most connected routers in the PoP are selected as RRs for redundancy purposes
  - All the routers in a PoP are iBGP clients of the two RRs in the PoP
  - Moreover, a full-mesh of iBGP sessions is configured between the RRs
- Both designs follow the recomendation in RFC 4456 [PelsserCN2010]

Zhang

- "Zhang" iBGP design
  - Define guidelines for hierarchical route-reflection in large Service Provider networks
- Recommendations
  - RRs at the top-level must be fully meshed
  - Mesh is not required for RRs at lower levels
  - Typically there are two levels of RRs (maybe more)
  - At the lowest level, the routers of a PoP are clients of the most connected routers of the PoP (as in "Bates2")
  - In turn, these RRs are clients of two RRs at the top-level
  - Finally, a full-mesh of iBGP sessions is configured between the top-level RRs

Bates & Zhang characteristics

Name	hierarchy	top-level full-mesh	PoP full-mesh	RR redundancy
Bates1	no	yes	yes	no
Bates2	no	yes	no	yes
Zhang	yes	yes	no	yes



Source: BGP Design and Implementation

- **Guidelines**
  - **Follow physical topology**
  - **Session between an RR and a nonclient should not traverse a client**
  - **Session between an RR and its client should not traverse a nonclient**

iBGP Correctness [GW02]

- Path symmetry: in eBGP signalling and forwarding traffic flow along the same path (usually peering over directly connected link)
- iBGP is routed, therefore path symmetry is not guaranteed

- iBGP configuration correctness: stable, anomaly free routing (in particular, loop-free)
- Checking the correctness of an iBGP graph is NP-complete
- Two conditions ensure a correct (loop-free) iBGP graph:
  - 1) route-reflectors should prefer client routes to non-client routes
  - 2) every shortest path should be a valid signaling path

Correct and scalable proposals

- BGPSep [Vutukuru2006how]
- Optimal iBGP topologies [BuobUM2008]
  - fm-optimality
- Skeleton [SarakbiM2010]

How to Construct a Correct and Scalable iBGP Configuration

- **IEEE INFOCOM 2006**
  - **Mythili Vutukuru**
  - **Paul Valiant, Swastik Kopparty and Hari Balakrishnan**

BGPSep contribution

- Status quo in configuring iBGP
  - Full-mesh (not scalable)
  - Route reflection (no correctness guarantees)
- Problems with both approaches
- New approach to configure iBGP that is both correct and scalable
- Uses results from graph theory

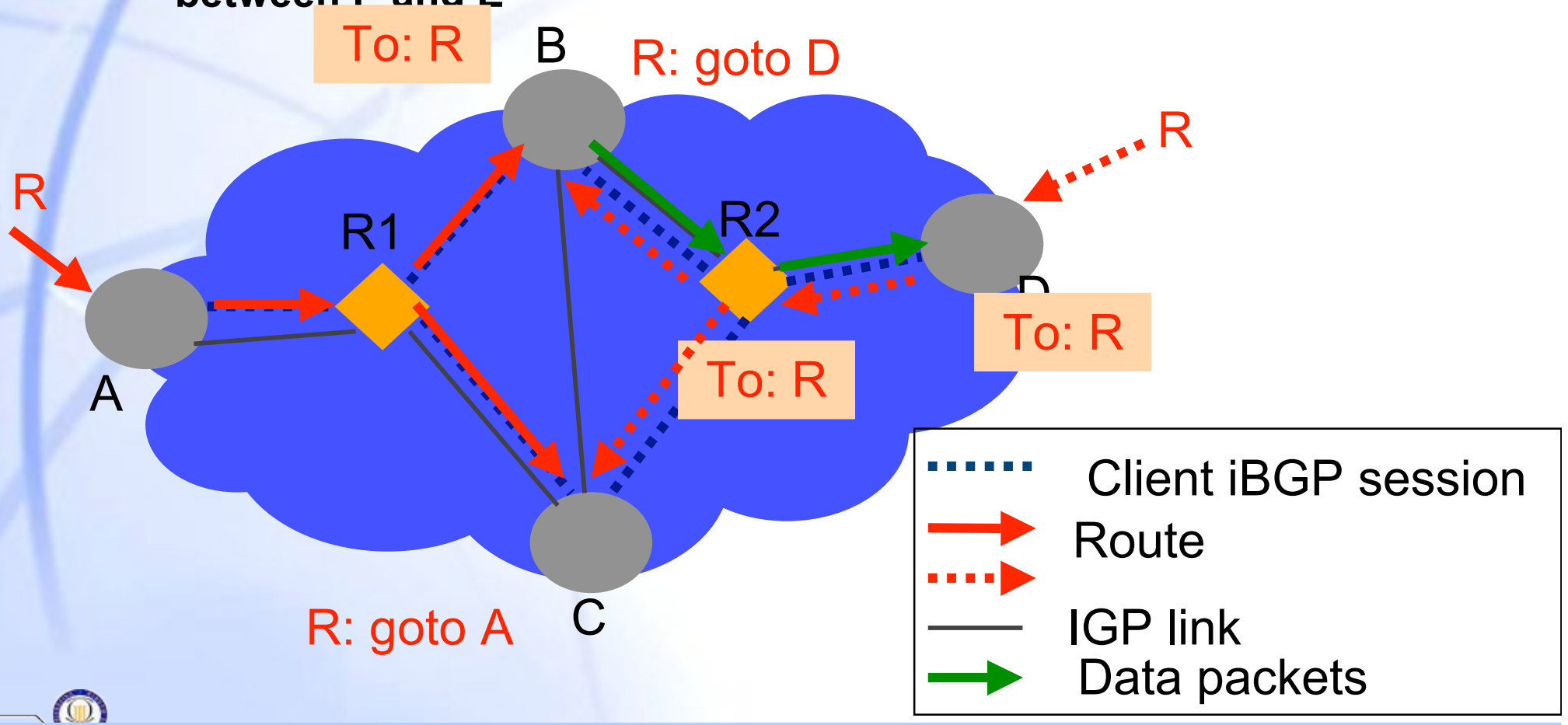
iBGP configuration	Correctness	Scalability
Full-mesh	√	×
Route reflection	×	√
BGPSep	√	√

BGSep: problem statement

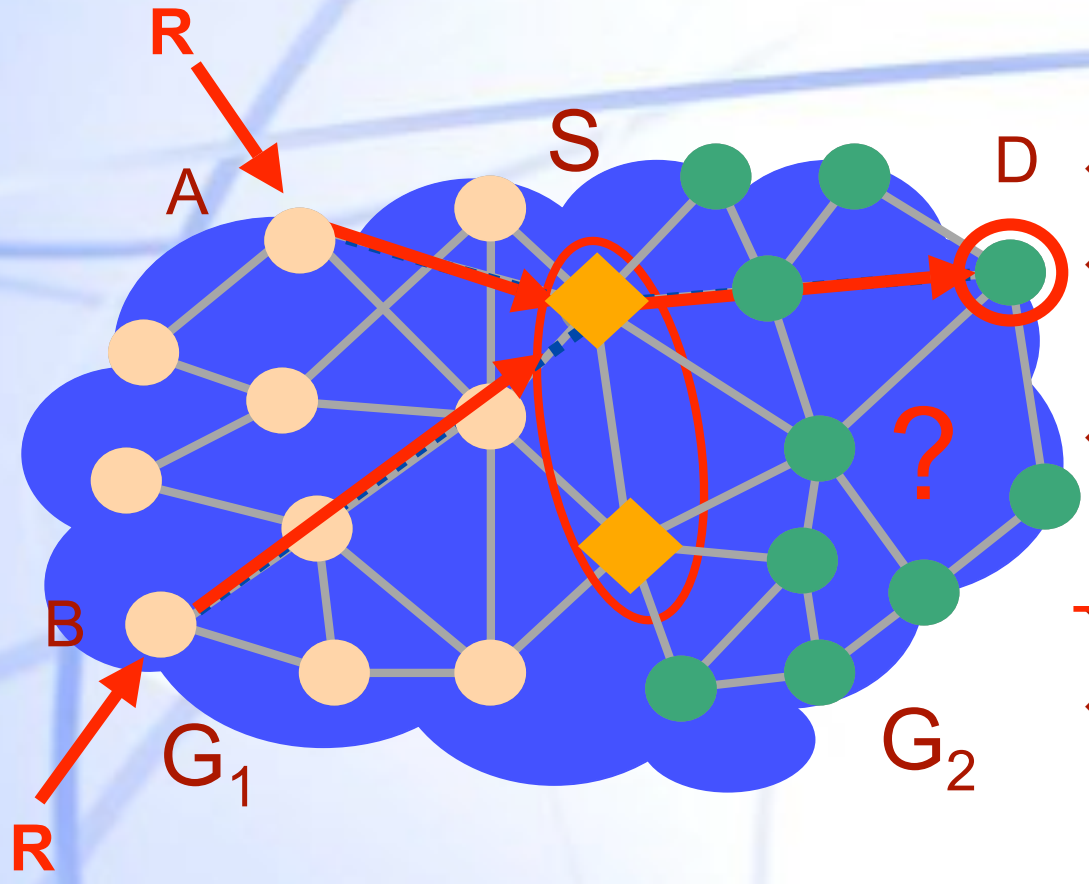
- **Input: IGP (IP-level connectivity) graph**
- **Output: iBGP configuration**
  - **Route reflectors and clients**
  - **iBGP sessions**
- **Constraints**
  - **Emulate full-mesh**
  - **More scalable than full-mesh**
- **Previous work [GW02] – how to *check* for correctness, not how to *construct* correct configurations**

Key insight for emulating full-mesh

- For every router P, every egress E
  - P and E have iBGP session, OR
  - P should be the client of a route reflector on the shortest path between P and E



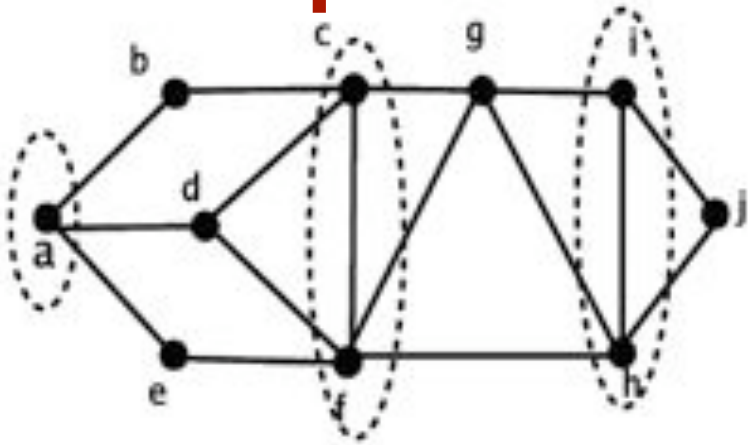
BGPSep solution



- S is graph separator
- Nodes in graph separator S are route reflectors
- u in G_1 or G_2, v in S: u is a client of v
- ~~Full-mesh in G_1, G_2~~
- Recurse on G_1, G_2

```
Algorithm BGPSep
Input: IGP Graph G, set V of BGP routers
Output: Set I of iBGP sessions
if |V| = 1 then
    I = ∅;
else if |V| = 2 then
    {u, v} ← V ;
    I = {(u, v, peer)};
else
    /* Step 1: Choose a graph
       separator S ⊆ V. Routers in
       S are the route
       reflectors.                 */
    S ← Graph-Separator(G) ;
    G_1, ..., G_m ← components of V − S;
    /* Step 2: Fully mesh the set
       of route reflectors         */
    foreach u, v ∈ S, u ≠ v do
        I = I ∪ {(u, v, peer)};
    end
    foreach G_i do
        /* Step 3: Make every
           router in each component
           G_i a route reflector
           client of every route
           reflector                */
        foreach u ∈ G_i, v ∈ S do
            I = I ∪ {(u, v, client)};
        end
        /* Step 4: Recursively
           apply BGPSep over each
           component                */
        I_i = BGPSep(G_i) ;
        I = I ∪ I_i ;
    end
end
return I;
```

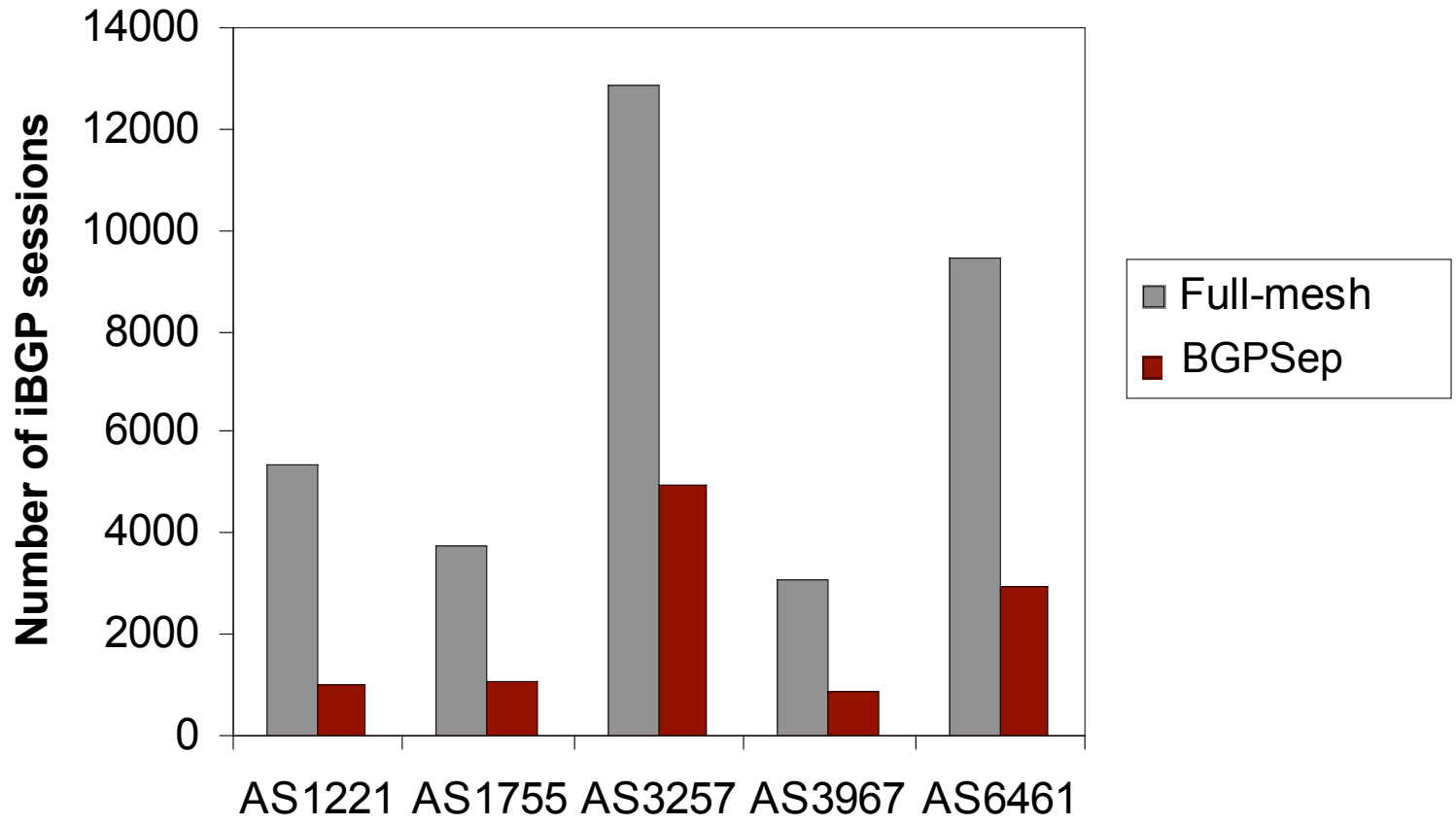
BGPsep algorithm and example



- Top level RR: {c,f}
- 2nd level RR: {i,h},{a}
- Clients: {b,d,e}, {g,j}
  - Sessions with top level **AND** 2nd level RRs
  - 25 iBGP sessions (45 iBGP sessions in FM)

Evaluation

- 2.5 to 5X fewer iBGP sessions on ISP topologies [Source: Rocketfuel]



Design optimal iBGP route-reflection topologies

- **IFIP Networking 2008**
  - **Marc-Olivier Buob – Orange Labs, LERIA**
  - **Steve Uhlig – Delft University of Technology**
  - **Mickaël Meulle – Orange Labs**

iBGP network design problem

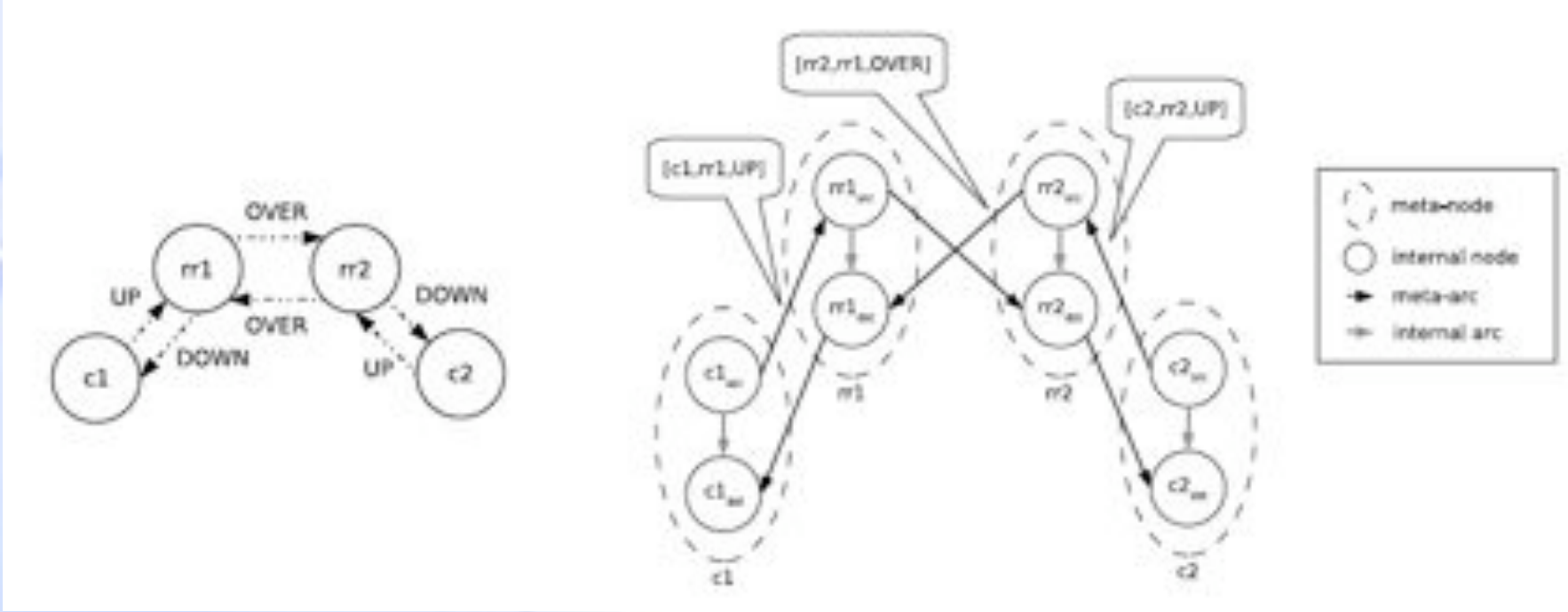
- **Inputs**
  - IGP topology V_{igp}
  - set of BGP routers "targets" ($R \subseteq V_{igp}$)
  - set of border routers "sources" ($N \subseteq R$)
- **Variables/Output**
  - iBGP topology
- **Constraints**
  - *fm-optimal* routing i.e. as in a full mesh topology with any set of concurrent border routers
    - ✓ implies a loop free and deterministic routing
  - *fm-optimal* routing even in case of a link failure
    - ✓ implies a loop free and deterministic routing even if a link fails
- **Objective**
  - iBGP topology should match as much as possible IGP topology
  - Less possible sessions

How to handle fm-optimality constraints?

- Give a sufficient condition to guaranty that a router r (in R) may be able to receive the route exiting at border router n (in N) when route entering at n is the best among all possible
  - a router w is *white* for (n,r) if it never blocks route propagation from n to r
  - IF a valid iBGP path composed of white routers exists between n and r THEN *r will learn its fm-optimal route from n*
  - condition works for any set of concurrent border router (only IGP weights needed)

How to handle fm-optimality constraints?

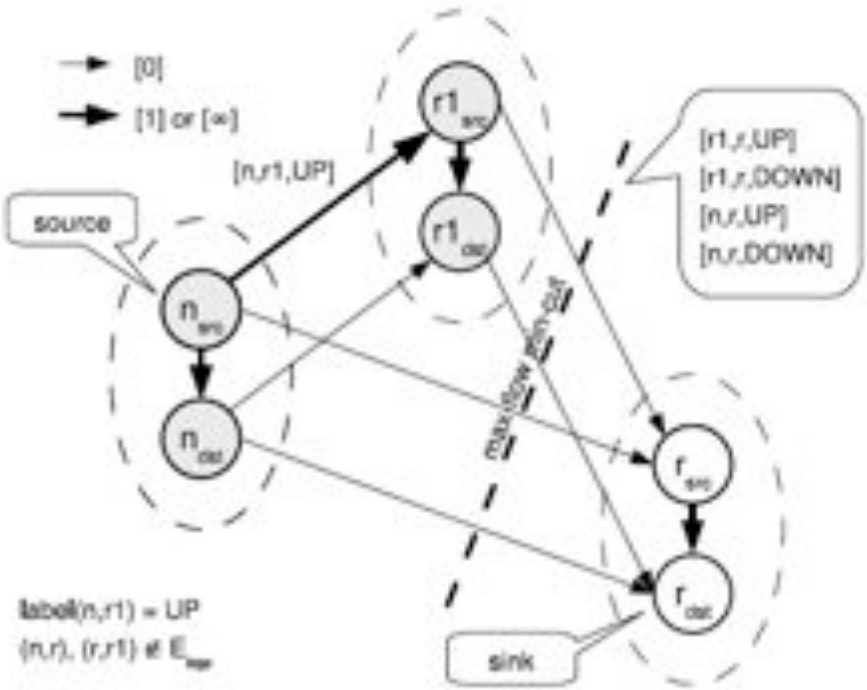
- **Use a graph transformation to easily look for valid and white iBGP paths**
  - **computing a valid iBGP path in a topology is equivalent to computing a normal path in the corresponding extended graph**



An example of iBGP graph and its corresponding extended graph. The valid path (c1, rr1, rr2, c2) in G_{bgp} is mapped to (c1src, rr1src, rr2src, rr2dst, c2dst) in G_{ext}

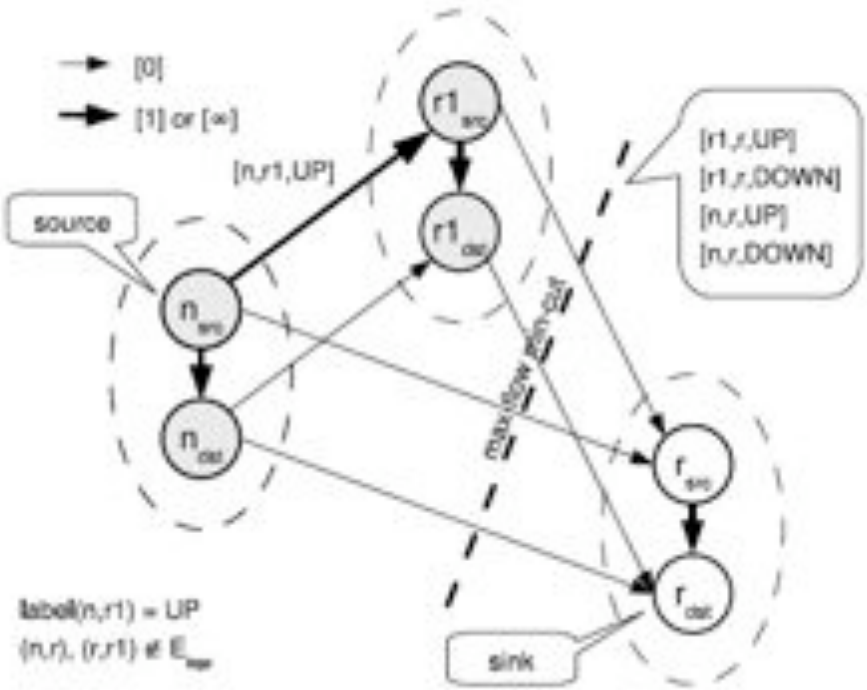
Benders decomposition: divide & conquer

- **Satellite problem for each (n,r) in (N,R)**
  - It looks for a white iBGP path in the extended graph for a (n,r) pair
  - A Flow problem is solved and outputs a new constraint if no such path exists
  - Max-flow Min cut, source n, sink r



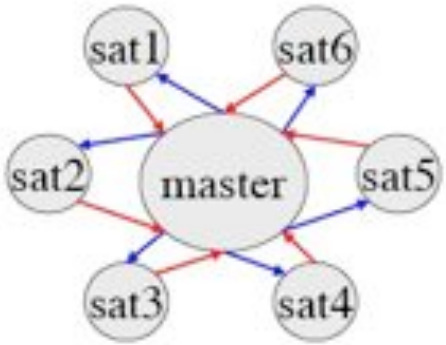
Benders decomposition: divide & conquer

- **Satellite problem for each (n,r) in (N,R) and each IGP link failure**
  - **Restrict iBGP sessions between routers in the same IGP connected component**
  - **Solve the same satellite as before**
- **Robust fm-optimality**
  - **= all satellite problems simultaneously satisfied**



Algorithm

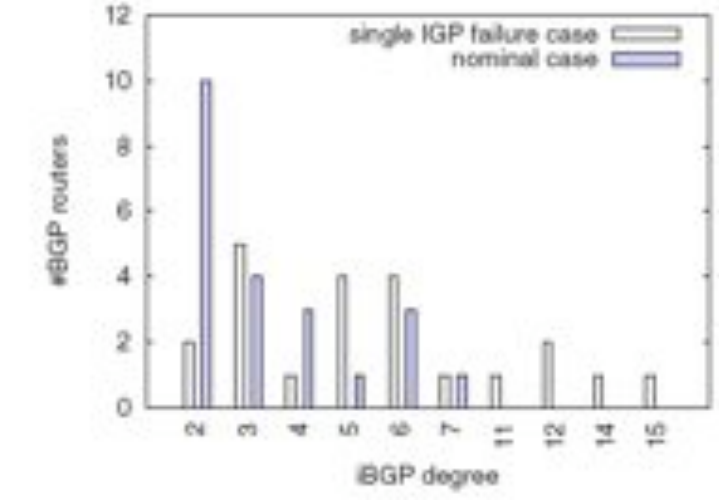
```
Do
        Solve master problem (Integer Linear Program)
        Inject solution found into satellites
        Interrogate satellites untill 1 or more unsatisfied
              Each unsatisfied satellite add a new constraint
              to the master problem
While at least one satellite unsatisfied
Return optimal solution
```

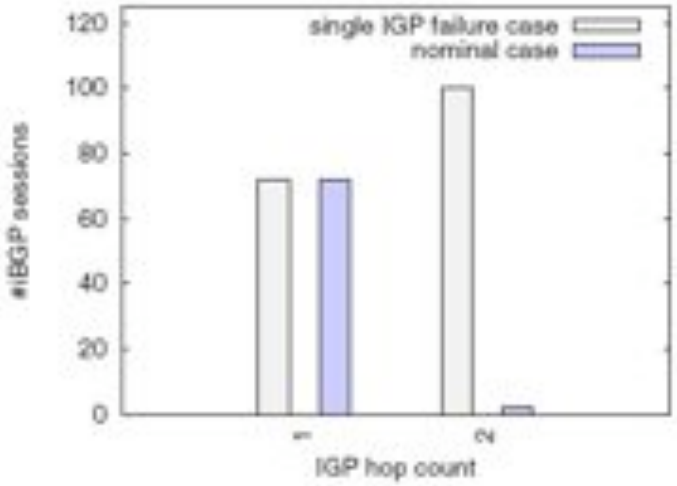


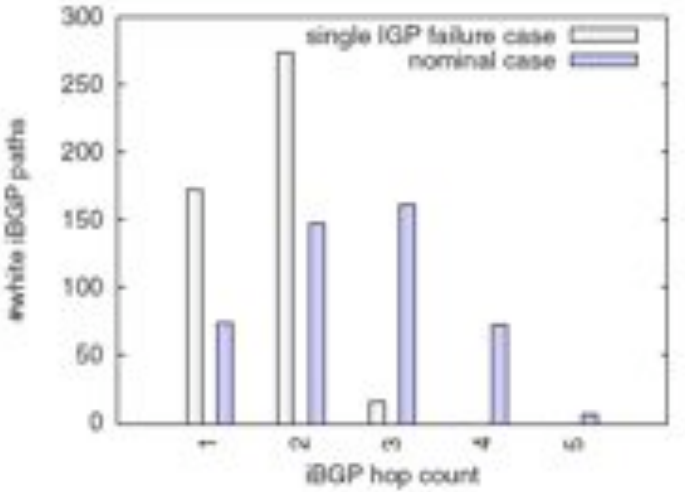
- **Objective function of the master problem gives incentive for:**
  - **the iBGP topology to follow IGP topology**
  - **Minimizing of number of sessions**

Results

- GEANT (22 nodes)







Characteristics of the solutions

- Not hierarchical, significantly different from real topology
- BUT realistic compared to the real topology:
  - iBGP paths with similar length (convergence)
  - Approx. 4x less iBGP sessions than FM in the robust case, 2 times less otherwise
- Very few multi-hops sessions
- Robust

BGP Skeleton - An Alternative to iBGP Route Reflection

- **IEEE INFOCOM 2010**
  - **Bakr SARAKBI and Stephane MAAG**
  - **Telecom SudParis**

Skeleton

- **Alternative to route reflection**
  - **Correct: it holds the sufficient correctness conditions as well as robustness against MED induced oscillations**
- **Skeleton**
  - **Subgraph of the physical graph with the same set of nodes**
  - **Its edges are the iBGP sessions between the nodes**
- ***Every Skeleton node is a Route Reflector***
  - **Skeleton eliminates the use of clusters and establishes iBGP sessions *only between single hop neighbors***
- **Number of iBGP sessions has a linear relationship with the number of ASBRs**
  - **Remember: in full-mesh this relationship is quadratic**

Basic Idea

- **Built over IGP neighbors only**
  - **Calculate the best IGP path between each internal node and each ASBR**
  - **Then an iBGP session is established between that node and its next-hop in the optimal path**

- **This method ensures that each node receives all the advertised prefixes from the optimal next hop to each ASBR, and hence all the nodes in the AS are able to determine their best exit point**

Algorithm

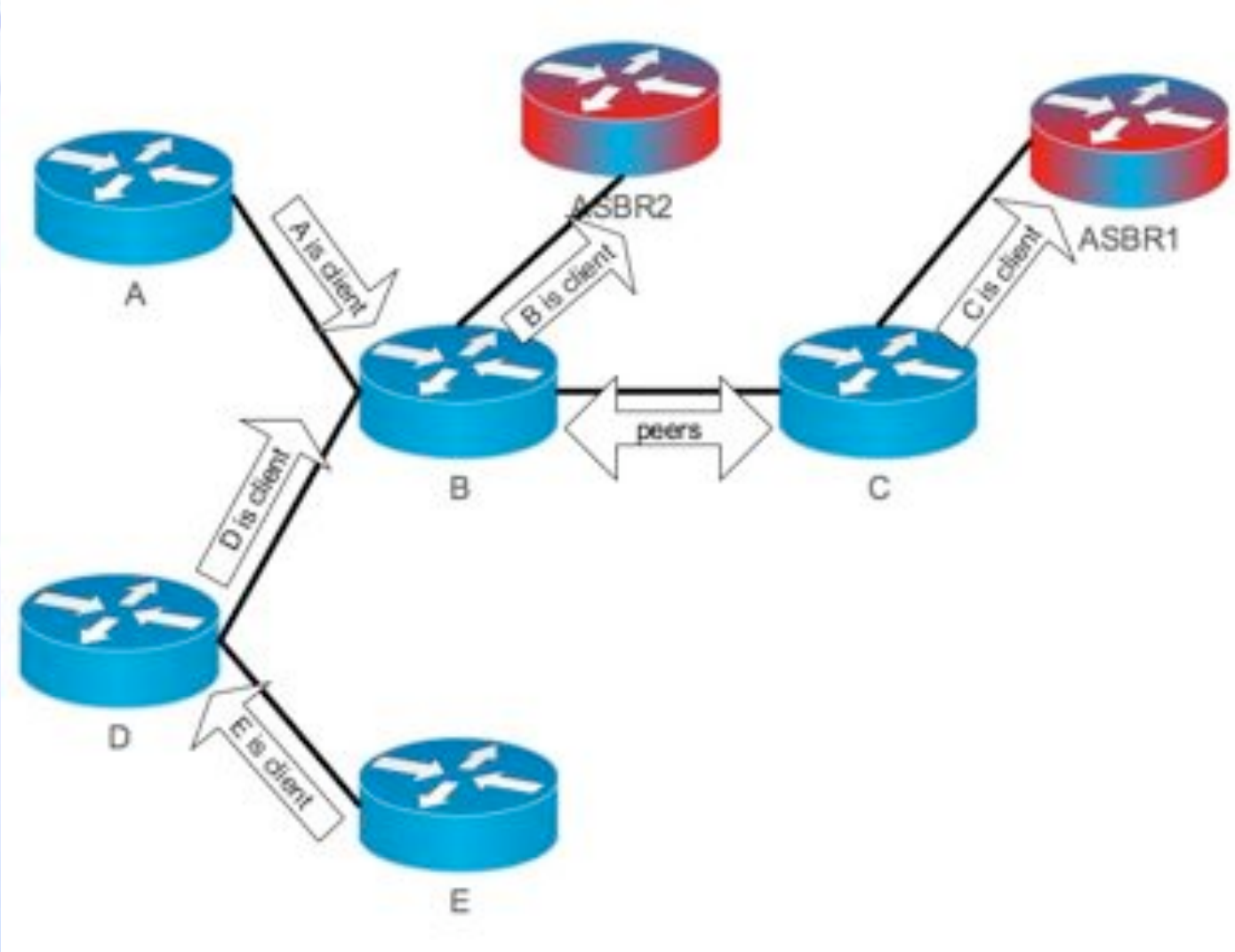
INPUT: Set of Nodes (V)∧ Set of ASBRs (Γ)
OUTPUT: Skeleton Subgraph $G_s(V,E_s)$

```
for each n ∈ V do
      for each α ∈ Γ do
            s ← βn(α)
            /* βn(α): is a function that gives the best
            IGP next hop to reach the ASBR α * /
            IBGP(n, s)
            /* IBGP: is a function that establishes an iBGP
            session between n and s * /
            Es ← Es U {n ←iBGP→ s}
            /* the new edge is added to the Skeleton
            subgraph * /
      end for
end for
```

Skeleton Session Types

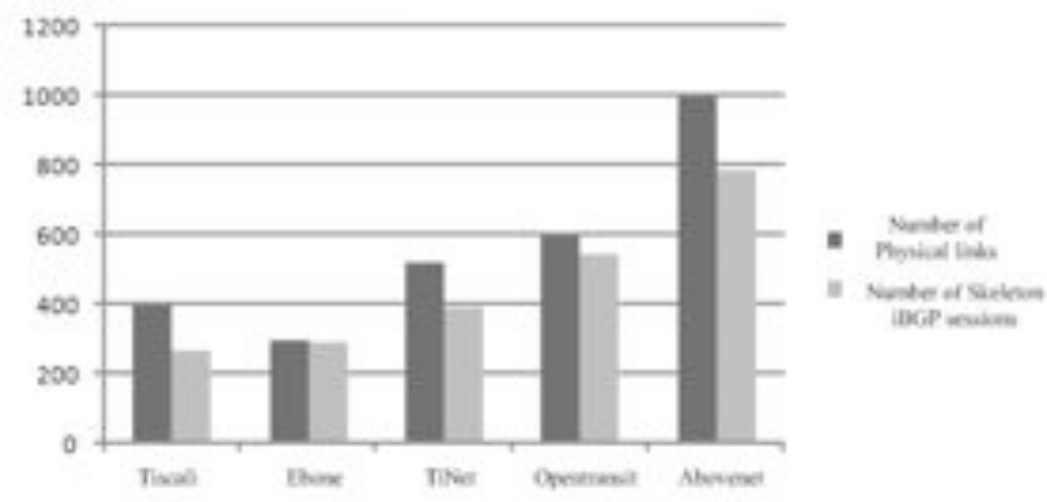
- The *successor* of router n_i in the best path towards ASBR α_j is router n_{i+1}, the preferred next hop for n_i to reach α_j
  - The successor reflects all its routes to its predecessor
    - ✓ The succesor is a Route Reflector for its predecessor
- n_i is a *predecessor* to n_{i+1} in the previous definition
  - The predecessor reflects its routes and routes of its predecessor to its successor
    - ✓ The predecessor is a client of its succesor, and a Route Reflector for its predecessor
- For two or more intersected paths in common nodes, if there exists (n,m) so that *n=successor(m)* for some ASBR and *m=successor(n)* for another ASBR, then *n=peer(m)*, equivalent to *m=peer(n)*
  - The two peers exchange all their routes

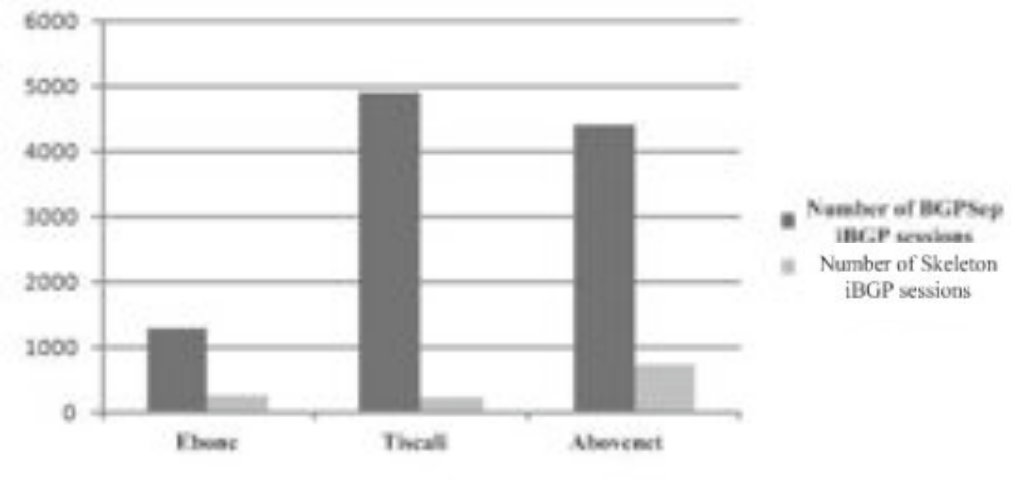
Skeleton Session Types



Results

Name	ASN	Internal Routers	ASBRs a	Links	Nb_{max}	Nb_{avg}
Tiscali	3257	276	161	400	13	4
Ebone	1755	163	87	300	16	8
TiNet	9121	325	125	525	15	3
Opentransit	5511	355	134	600	12	4
Abovenet	6461	367	138	1000	25	7





iBGP design algorithms

- **Bates & Zhang heuristics**
  - Widely deployed in Service Provider networks
  - Correcteness is not assured
    - ✓ Configuration errors may trigger instabilities
- **BGSep, Optimal, Skeleton**
  - Key issues: correctness AND scalability
  - NO changes to regular BGP
  - Solve design problem
  - Prove correctness
  - Promising comparison to FM

References

1. [RFC 4456] T. Bates, E. Chen, R. Chandra, *BGP route reflection - an alternative to full mesh internal BGP (IBGP)*, RFC 4456 (April 2006).
2. [PelsserCN2010] C. Pelsser, B. Quoitin, S. Uhlig, T. Takeda, and K. Shiomoto. *Providing scalable NH-diverse iBGP route redistribution to achieve sub-second switch-over time*. To appear in Computer Networks journal, 2010.
3. [BGPDesign] R. Zhang, M. Bartell, *BGP Design and Implementation*, 1st Edition, Cisco Press, 2003.

References

4. [GW02] Griffin, T.G.,Wilfong, G.: *On the correctness of iBGP configuration*. In: Proc. of ACM SIGCOMM (August 2002).
5. [Vutukuru2006how] *How to Construct a Correct and Scalable iBGP Configuration*, Mythili Vutukuru, Paul Valiant, Swastik Kopparty, and Hari Balakrishnan. IEEE INFOCOM 2006.
6. [SarakbiM2010] *BGP Skeleton -An Alternative to iBGP Route Reflection*, Bakr SARAKBI, Stephane MAAG. IEEE INFOCOM 2010.
7. [BuobUM2008] M.-O. Buob, S. Uhlig, M. Meulle, *Designing optimal iBGP Route-Reflection topologies*. IFIP Networking 2008.