Checking for optimal egress points in iBGP routing

Marc-Olivier Buob
France Telecom R&D / LERIA
marcolivier.buob@orange-ftgroup.com

Mickael Meulle
France Telecom R&D
michael.meulle@orange-ftgroup.com

Steve Uhlig
Delft University of Technology
S.P.W.G.Uhlig@ewi.tudelft.nl

Abstract— **The Border Gateway Protocol (BGP) is used today by routers of all Autonomous Systems (AS) in the Internet. BGP is responsible for end-to-end reachability in the Internet. BGP routers have to exchange routes towards about 200,000 prefixes (blocks of IP addresses). Inside each AS, routers are using internal iBGP sessions to exchange their best route with other routers. The objective of an iBGP topology is to redistribute BGP routes inside the whole AS. In large ASs, some BGP routers are used as route reflectors because it reduces the total number of sessions needed for this route redistribution.**

Checking the correctness of iBGP configuration [1] and detecting potential problems inside the iBGP [2] are particularly difficult when route reflection [3] is used. The scalability of route reflection compared to an iBGP full-mesh comes at the cost of opacity in the choice of the best routes by the routers inside the AS. This opacity induces problems like suboptimal route choices in the IGP cost, deflection and forwarding loops.

We propose to check for the optimality of the routes choice in a route reflection graph. We do so without requiring to simulate the complex operations of the BGP protocol. Our check procedure is applied to a tier-1 AS. Our simulations show that a significant fraction of suboptimal path choices may occur, between 10 and 30% in realistic cases.

I. Introduction

The Internet consists of a collection of more than 21,000 domains called Autonomous Systems (ASs). Each AS is composed of multiple networks operated under the same authority. Inside a single domain, an independent Interior Gateway Protocol (IGP) [4] such as IS-IS or OSPF is used to propagate routing information. Between ASs, an Exterior Gateway Protocol (EGP) is used to exchange reachability information. Today, BGP [4] is the de facto standard interdomain routing protocol used in the Internet. BGP routers exchange routing information over BGP sessions. External BGP (eBGP) sessions are established over inter-domain links, i.e. links between two different ASs (BGP peers), while internal BGP (iBGP) sessions are established between the routers inside an AS. Through its BGP sessions, each router receives and propagates BGP routes for destination prefixes. A BGP router processes and generates route advertisements as shown in Figure 1. Administrators specify input filters per BGP peer, which are used to discard unacceptable incoming BGP advertisements. Once a route advertisement is accepted by the input filter, it is placed together with the routes originated at this router in the incoming Routing Information Base (RIB-In). Beforehand, some of the route attributes may have been modified according to the local routing policies. Next, the BGP decision process is used to select the *best route* for each prefix among the available routes. This route is then placed into the BGP routing table, which we will also refer to as the *RIB-Out*. Finally, administrators may specify output filters for each peer, which are used to decide which best routes to propagate to a BGP neighbor.

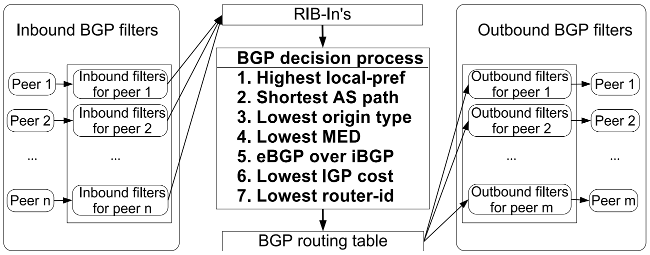

Fig. 1. Operation of a BGP router.

The BGP decision process consists of a sequence of elimination steps. Its final goal is to select a single best route for any given prefix. For this purpose, the BGP decision process considers several of the BGP routes attributes. One of the first attributes is `local-preference` (in short, `local-pref`). As `local-pref` is a non-transitive attribute, it can be used to locally rank routes. The next BGP attribute examined by the BGP decision process is the `AS-path`. An `AS-path` contains the sequence of ASs that a route crossed to reach the current AS. Routes with shorter `AS-paths` are preferred. Next in the evaluation process is the `multi-exit-discriminator` (`MED`). This attribute is used to rank routes received from the same neighbor AS, but it can also be used across neighbors. Then the decision process ranks routes according to the IGP cost of the intra-domain path towards the exit point in the AS (called the BGP next-hop), preferring routes with smaller IGP cost. This rule implements hot-potato routing [5]. Finally, if there is still more than a single route left, the router breaks ties, for example by selecting the route to the neighbor which has the lowest `router-ID` (typically one of its IP addresses).

A. iBGP route reflection

Route-reflection [3] was initially introduced as an alternative to the iBGP full-mesh that requires $n(n-1)/2$ iBGP sessions to be established inside an AS. This number of sessions required for propagating the routes learned from neighboring ASs does not scale for large networks containing hundreds or thousands of BGP routers. Route-reflection was thus introduced to limit the number of iBGP sessions for large size networks. An advantage of an iBGP full-mesh is that all routers know the best routes of the other routers inside the network. When a route is withdrawn, routers can typically switch to another route immediately, without waiting for BGP to converge. When an iBGP full-mesh is not used, routers might know only a very limited subset of the routes the AS knows to reach an external destination.

Route-reflection inside an AS defines two types of relationships among BGP routers: client and non-client. These relationships among BGP peers define a loose hierarchy among routers, going from the bottom level routers that have no clients up to the largest route reflectors that are not client of any other router. Note that this implicit hierarchy is not practically enforced, as iBGP sessions can be established between any two routers inside the AS, even under route reflection.

The redistribution of the routes in iBGP works according to well-defined rules. First, recall that a route is never re-advertised to the peer that announced it. Consider a given prefix p for which a router inside the iBGP receives several routes from its peers (iBGP or eBGP). The router chooses its best route towards p among the possible ones, using the BGP decision process [6]. How the best route is propagated to the iBGP neighbors of a router depends on whether the router acts as a router-reflector or not. If a router does not act as a route reflector, i.e. it has no "client" peer, then the router advertises this route to all its iBGP peers if it is learned from an eBGP session, or to none of them if the route was learned from an iBGP session. On the other hand, if a router acts as a route reflector [3]:

- If the route was learned from a client peer (or eBGP peer), the route reflector redistributes the route to all its clients and non-client peers (except the one from which the route was received).
- If the route was learned from a non-client peer, the route reflector redistributes the route to its client peers only.

These rules driving the redistribution of the routes inside the iBGP imply a filtering of the routes over the internal BGP signaling graph. Besides the rules defined in [3] when connecting route reflectors to ensure a proper working of the iBGP propagation, there is no clear design rules known today as to how to design a proper iBGP graph. Guidelines for checking that a correct iBGP configuration have been discussed in [1]. [2] provides a tool to detect potential problems due to the iBGP configuration based on static analysis.

B. Drawbacks of route reflection

The main drawback of route reflection is the potential lack of visibility of some routes. A client router of a route reflector typically does not know the same routes as it would have known in a full-mesh. Each router may not know all routes needed to select the best path among all possible ones known inside the AS. The use of route reflection limits route diversity in the network [7].

Suppose that for a given prefix, a route reflector knows multiple BGP routes distinguished on the IGP cost to their BGP next-hop. This route reflector will only forward its route to its closest BGP next-hop. Depending on the choice of the route reflectors, some BGP speakers may be unable to learn (and thus to choose) their own best route in term of the IGP cost. A such loss of information may lead to many routing problems as explained below.

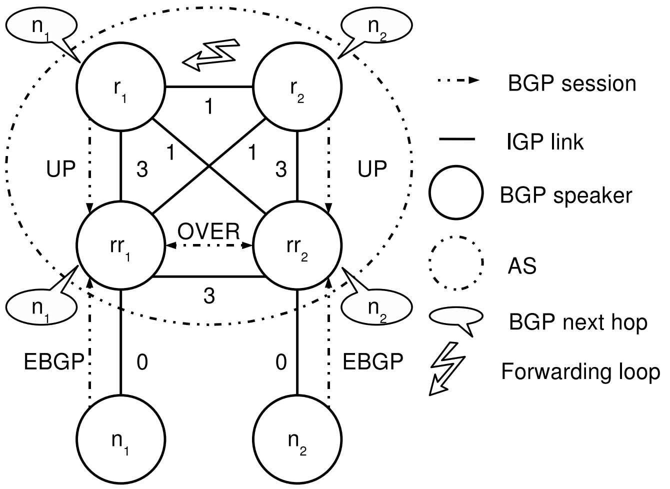

Fig. 2. Packets sent from r_1 and r_2 are trapped in a forwarding loop $r_1, r_2, r_1, r_2, ...$

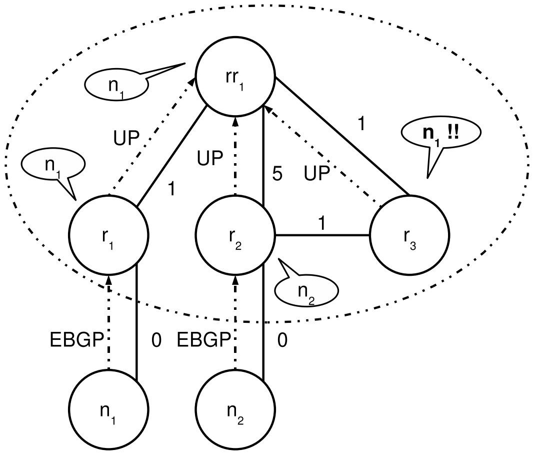

Fig. 3. r_3 selects the route announced by n_1 because it never learns the route announced by n_2.

- **Suboptimal egress point:** Lack of visibility induced by route reflectors might lead to suboptimal routing [8]. A BGP speaker will not always learn the route exiting to its closest next-hop due to route reflection. Figure 3 shows an example of suboptimal routing. Because rr_1 prefers the route announced by n_1, r_3 never learns the route announced by n_2 and thus cannot send its traffic to its closest egress point.
- **Non-determinism:** Depending on the order in which BGP messages arrive, the best route selected by some routers might differ [1]. The network state is then not predictable. Let's consider the example reported in Figure 4. For a given prefix p, let be ρ_i the route announced by $n_i, i \in \{1, 2\}$

1) Suppose that the considered AS learns ρ_1 at first.

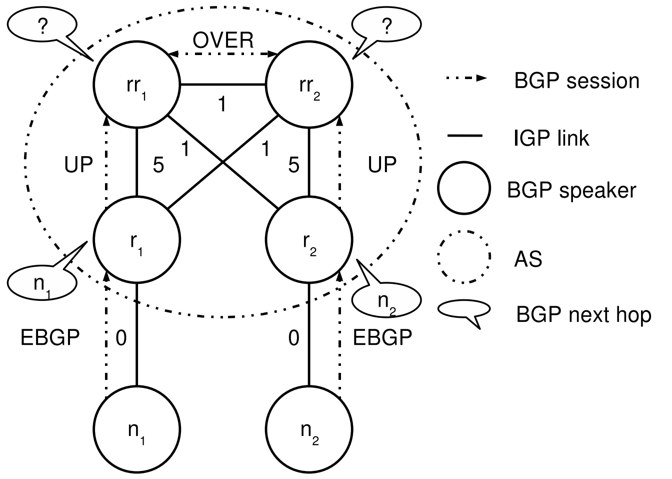

Fig. 4. Two possible final states: rr_1 and rr_2 both select the route learned by n_1 or n_2.

Each router selects route ρ_1 for this prefix p. Then, the considered AS learns ρ_2. r_2 selects ρ_2, but rr_2 keeps its current route ρ_1. The final stable state is $S_1 = \{r_1 : \rho_1; rr_1 : \rho_1; rr_2 : \rho_1; r2 : \rho_2\}$.

2) Now, suppose that the AS first learns ρ_2. By the same reasoning, routing converges to the final state $S_2 = \{r_1 : \rho_2; rr_2 : \rho_2; rr_2 : \rho_2; r2 : \rho_1\}$, and $S_1 \neq S_2$. This topology hence does not lead to deterministic routing.

- **Forwarding loops caused by deflection:** Suppose two BGP routers have selected distinct BGP best routes towards a prefix that are tie-breaked by the IGP cost to the next-hop. If a BGP speaker selects a BGP next-hop, and if the IGP path from this router to this next-hop reaches a BGP speaker that selects another egress point, the traffic is said to be *deflected*. If multiple deflections occur, forwarding loops can appear [8]. For example, in Figure 2, the same prefix is advertised to two BGP routers via eBGP sessions. rr_1 and rr_2 always select the eBGP-learned route as best for the prefix. r_1 selects the route advertised by rr_1 with n_1 exit point, and r_2 selects the route advertised by rr_2 with n_2 exit point. r_1 is on the IGP path from r_2 to n_2 and r_2 is on the IGP path from r_1 to n_1. There is thus a forwarding loop between r_1 and r_2.

The convergence of the BGP protocol occurs independently for each prefix [1]. Since BGP updates are packed in messages, and since messages are sent every x unit of time, convergence of the network is slow when too many prefixes are updated at the same time [9]. For any prefix, some of the problems mentioned above can occur. The configuration of the routers in an AS follows the specifications and guidelines initially decided by the network managers. In practice however, complex interactions between IGP and BGP can happen with the use of route reflector hierarchies [10], [11]. Furthermore, when internal and external network failures occur, the network can fall into dangerous states where many other prefixes will experience iBGP convergence problems [12].

C. Checking for optimal routes propagated by the iBGP configuration of an AS

To our knowledge, no work so far has proposed a method to check if a BGP router selects the route towards its closest egress point, as would be the case in an iBGP full-mesh. We call this kind of optimality *fm-optimality* (fm as in full-mesh).

When tie-breaking of some BGP routes for the same prefix is done on the IGP cost to the next-hop, it is possible that deflection occurs for some routers. Because routing deflections limit the predictability of network state and can alter routing and forwarding performance (loops, optimal routing . . .), we believe that a network should experience a minimal number of routing defections. Moreover, when packets are deflected by a router, the alternate BGP route used to exit the AS through a new next-hop can be significantly different from the original route selected for the packets. This route may indicate an alternate entry point in the same neighbor AS, or another neighbor AS with a completely different `AS-path`. The unpredicted new route taken by packets may not be the one that matches the goals of the network managers.

We show in this paper how to check for our so called *fm-optimality*. We use as input the IGP topology and the iBGP topology of an AS with the set of possible BGP next-hops. Our methodology performs a complete and systematic verification of deflections that would occur between a BGP router r and any border router of the AS that would have been one of r's closest exit points for some prefixes. Sets of possible concurrent exit points for each prefix can be used as input. For the sake of generality, we consider all BGP next-hops to be concurrent for any given destination network. In practice however, one may want to restrict for each prefix the set of next-hops to those from which routes towards this prefix may be received.

Our *fm-optimality* check procedure is applied to a large tier-1 AS and reveals up to 30% of potentially suboptimal routes.

The rest of this paper is organized as follows. In section II we introduce our notations and network models for IGP and iBGP routing. In section III, we explain our methodology to check for *fm-optimality* in the network. In section IV, we show results for a tier-1 AS. Finally we conclude and discuss further work.

II. Network model

To model the propagation of BGP routes within an AS, we use one graph for IGP routing and one graph for BGP sessions. In the IGP graph, paths are selected by routers using shortest paths. In the BGP reflection graph, the propagation of routes follows the rules of iBGP route reflection. We introduce at the end of this section a topological model to compute valid iBGP signaling paths inside the AS. The selection of a best BGP route follows the BGP decision process [6], and decides which routes are propagated by routers. In this paper we study the cases where tie-break occurs at the step of comparing the IGP cost towards the next hops (see part III).

[1] The BGP decision process of a router is applied separately to two distinct prefixes when aggregation is disabled or not applicable.

A. Graph models

1) "IGP Graph" (intra-domain routing): Let be $G_{igp} = (V_{igp}, E_{igp})$ the directed graph for the IGP topology. V_{igp} is the set of routers and external BGP next-hops. E_{igp} is the set of arcs between routers in V_{igp}. We add two arcs $e = \overrightarrow{(u,v)}, e' = \overrightarrow{(v,u)} \in E_{igp}$ when the two routers $u, v \in V_{igp}$ share a link. Each edge is weighted with the IGP metric $w\overrightarrow{(u,v)}$ from u to v, and e' edge is weighted with the IGP metric $w\overrightarrow{(v,u)}$ from v to u. In real network topologies, a BGP next-hop address has to be reachable via IGP routing for packet forwarding to be possible. Peer routers connected to an AS with an eBGP session are not systematically in the IGP topology of the AS (for example when static routes are defined). When a BGP next-hop is not in the IGP network topology, we add a router labeled with the IP address of the next-hop. We also add to E_{ibgp} the arc from the border router of the AS that establishes the eBGP session to this next-hop. We set its IGP metric to 0.

We denote by $dist : V_{igp} \times V_{igp} \longrightarrow \mathbb{N}$ the weight of the shortest path between two routers.

2) "Reflection graph" (BGP route redistribution) : We denote by $\mathcal{N}$ the set of BGP next-hops, and $\mathcal{R}$ the set of routers running BGP inside the AS. The reflection graph $G_{bgp} = (V_{bgp}, E_{bgp})$ describes the route reflector hierarchy. We have $V_{bgp} = \mathcal{R} \cup \mathcal{N}$. E_{bgp} denotes the set of BGP sessions between routers. When two routers share an iBGP session we add two edges between routers labeled with UP from a client to one of its route reflector, $DOWN$ from a route reflector to one of its clients, or $OVER$ (see Figure 6). eBGP links are labeled $EBGP$. We use the same notations as [8], [13].

We denote by $\mathbb{D} = \{EBGP, UP, OVER, DOWN\}$ the set of iBGP sessions type and $label : E_{bgp} \longrightarrow \mathbb{D}$ the label of a given link. We also note $sym : \mathbb{D} \longrightarrow \mathbb{D}$ the function that return the symmetric label of a given label. $sym(EBGP) = EBGP$, $sym(UP) = DOWN$, $sym(DOWN) = UP$, $sym(OVER) = OVER$.

3) Consistency of BGP sessions: We assume BGP sessions in the reflection graph to be possible and correctly established between two BGP peers:

$$\forall \overrightarrow{(u,v)} \in E_{bgp}, \begin{cases} \exists \text{ path } \mu \text{ in graph } G_{igp} \text{ from } u \text{ to } v, \\ \exists \text{ path } \mu' \text{ in graph } G_{igp} \text{ from } v \text{ to } u, \\ \overrightarrow{(v,u)} \in E_{bgp}, \\ label\overrightarrow{(u,v)} = sym(label\overrightarrow{(v,u)}). \end{cases} \tag{1}$$

To simplify our figures, we only draw one UP edge from a client router to a route reflector instead of drawing the two symmetric edges labeled with UP and $DOWN$.

4) Assumptions on BGP route filtering: In practice, BGP filters are deployed in eBGP sessions. We assume that:

- if a filter is set between a BGP next-hop n and a router r for the considered prefix p, we do not build the edge from n to r ;
- no filtering happens between two iBGP peers.

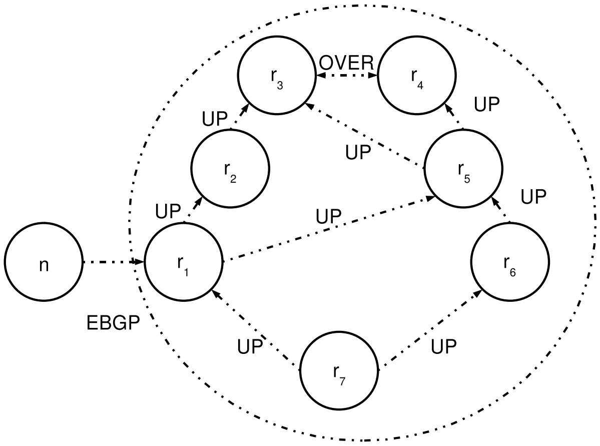

Fig. 5. Paths $(n, r_1, r_2, r_3, r_4, r_5, r_6)$, $(n, r_1, r_2, r_3, r_5, r_6)$, (n, r_1, r_5, r_6) are some valid paths in G_{bgp} from next-hop n to router r_6.

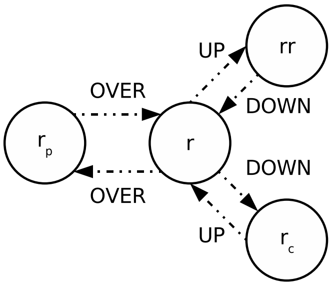

Fig. 6. A graph G_{bgp} with the three types of iBGP links.

B. Extended graph model for route propagation in reflection graph

Valid path, valid graph: A *valid path* $(r_1, ..., r_k)$ in the reflection graph G_{bgp} consists of zero or one $EBGP$ edge, followed by zero or more UP edge(s), followed by zero or one $OVER$ edge, followed by zero or more $DOWN$ edge(s) [14]. This pattern of path label can be described with the following regular expression:

$$(EBGP)?(UP)^*(OVER)?(DOWN)^* \tag{2}$$

Several examples of valid paths are reported in Figure 5.

G_{bgp} is a *valid graph* if and only if for each origin-destination pair $(n, r) \in \mathcal{N} \times \mathcal{R}$, a valid iBGP path from n to r exists.

Extended graph G^{ext}_{bgp}: To ensure the validity of each signaling path, we transform G_{bgp} into the G^{ext}_{bgp} graph according to Figures 6 and 7. Vertices of set $\mathcal{N}$ and eBGP links are not modified. The extended grah ensures that only paths that comply with the regular expression that defines a valid path are allowed [15].

III. CHECKING *fm-optimality* OF IBGP ROUTES

A. Definitions

Equivalent routes: Let us consider a valid and stable BGP topology. For a given prefix, if a route is better than others according a higher `local-pref` value, a shorter `AS-path`, lowest `origin-type` or lower `MED`, this route is chosen as best by all BGP routers in the AS. In this case, the iBGP

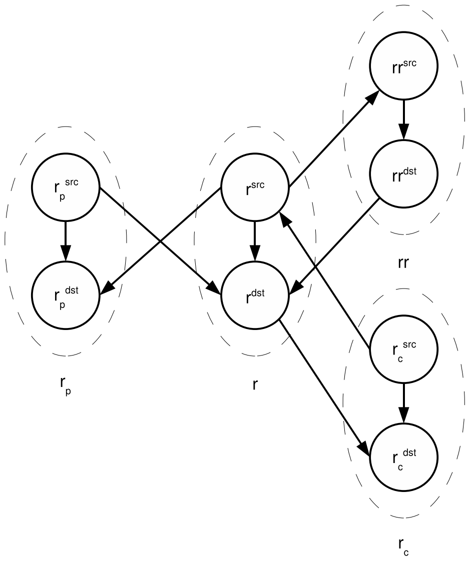

Fig. 7. The extended graph G_{bgp}^{ext} associated to G_{bgp} described by Figure 6.

topology leads to the same routing as a full-mesh of iBGP sessions. If routes toward the prefix cannot be distinguished with the `local-pref`, `AS-path`, `origin-type` or `MED` attributes, routes are said to be *quasi-equivalent*. In this case, the way in which router will select their best path depends on what is called *hot-potato routing*.

Two steps in the BGP decision process are responsible for hot-potato routing in the AS:

- routes learned by eBGP are preferred over routes learned by iBGP,
- routes to closest BGP next-hops in the IGP topology are preferred.

Routers of the AS select *quasi-equivalent* routes towards a prefix depending on their relative position (in terms of IGP cost) to the BGP next-hop of the routes.

Concurrent next-hops: We call *concurrent next-hops* the next-hops of *quasi-equivalent* routes towards a given prefix p. We denote this set by $\mathcal{CN}(p) \subseteq \mathcal{N}$.

Fm-next-hops: We call a *fm-next-hop* of a router r one of the BGP next-hops it would have selected after the two hot-potato steps of the BGP Decision process if the iBGP topology was a full-mesh.

Because we assume that IGP weights on inter-AS links are set to 0, any *fm-next-hop* n of a router r always verifies $dist(r, n) \leq dist(r, n')$ for any other *concurrent next-hop* n'.

Shadowing-reflectors: In the case of route reflection, *quasi-equivalent* routes with the *fm-next-hop* of some routers may be hidden by some route reflectors that select a route with another next-hop. Such route reflectors that hide *quasi-equivalent* routes are said to be *shadowing-reflectors*. For example, in Figure 3, rr_1 is a shadowing-reflector of pair (n_2, r_3) because rr_1 does not advertise the route with next-hop n_2 due to its preference for the route with next-hop n_1. Such shadowing reflectors hide *quasi-equivalent* routes whose next-hop is a *fm-next-hop* for some other routers. To be a *shadowing-reflector*, a route reflector rr must hence select as best a route ρ with next-hop n, and learn another route ρ' whose next-hop n' is a *fm-next-hop* for another router.

When a router r' is a shadowing-reflector for the pair (n, r), router r cannot learn routes with next-hop n through any valid signaling path $(n, .., r', ..., r)$. Let us denote by $\mathcal{S}(n, r) \subseteq \mathcal{R} \backslash \{r\}$ the set of shadowing-reflectors for router r to one of its *fm-next-hops* n. r learns a route with next-hop n if and only if a valid signaling path from n to r exists in graph G_{bgp} after pruning all shadowing-reflectors of the set $\mathcal{S}(n, r)$. This condition is said to be the *fm-optimality* condition for the pair (n, r).

- **Fm-optimal path:** a valid iBGP path in the reflection graph $(n, r_1, ...r_k, r)$ from a *fm-next-hop* $n \in \mathcal{N}$ to a router $r_k \in \mathcal{R}$ is said *fm-optimal* for (n, r) if and only if there is no *shadowing-reflector* in set $\{r_i\}_{1 \leq i \leq k}$.
- **Fm-optimal pair:** a pair composed by a router r and one of its *fm-next-hops* n is *fm-optimal* if and only if at least one *fm-optimal path* exists from n to r.

Fm-optimality of any pair is a strong condition for an iBGP reflection topology. But this condition leads to a reliable iBGP topology in the AS where the drawbacks of deflection can not occur (see section I-B). If the BGP topology is *fm-optimal* when all the BGP nexthops announce quasi-equivalent routes (i.e. $\mathcal{CN} = \mathcal{N}$), we can show that this topology is *fm-optimal* for all subset of $\mathcal{N}$. In this case, BGP topology verify the following properties:

- **Validity**: A BGP *fm-optimal* topology is also valid because each router learns at least the route to its closest nexthop.
- **Optimality**: *Fm-optimality* implies optimal routing because each router knows its closest egress point. In practice some (next-hop,router) pairs may not be *fm-optimal* while routing is still optimal. It depends on the different subsets of BGP next-hops announcing a given prefix.
- **Determinism**: Each router picks the route related to its closest next-hop. If all (router,egress point) pairs have a different IGP metric, BGP routing is deterministic.
- **No deflections**: If a router has only one closest egress point for a given prefix, and if no static route is defined on a router belonging to the IGP path from this router to its egress point, no deflexion occurs. Let us consider $(n, r) \in \mathcal{N} \times \mathcal{R}$, $r' \in \mathcal{R}$ such that r' belongs to the IGP path from n to r, and $n' \in \mathcal{N} \backslash \{n\}$ such that $dist(r, n) < dist(r, n')$. We have $dist(r, n) = dist(r, r') + dist(r', n)$ because r' belongs to the shortest path from r to n. Furthermore, $dist(r, n') \leq dist(r, r') + dist(r', n')$. Those two inequalities lead to $dist(r', n) < dist(r', n')$. r' prefers n to n'. Thus, traffic sent from r to n is never deflected by r'. Deflections may only occur when r' verifies $dist(r', n) = dist(r', n')$.
- **No forwarding loops**: If the IGP topology does not contain any zero-cost circuits, and if static routes may only be defined between a border router and a BGP

nexthop, we can prove that no forwarding loop can occur. Consider a forwarding loop due to multiple deflections. We denote by $r_1, \ldots, r_k \in \mathcal{R}$ the BGP speakers deflecting the traffic and $n_1, \ldots, n_k \in \mathcal{N}$ their respective BGP next-hops. Indexes are given modulo k. Each r_{i+1} forwards the traffic sent by r_i to n_i and redirects it to n_{i+1}. Each r_{i+1} belongs to the shortest path from r_i to n_i, thus $\forall i \in \{0, ...k\}$, $dist(r_i, n_i) = dist(r_i, r_{i+1}) + dist(r_{i+1}, n_i)$. Because the BGP topology is *fm-optimal*, each r_i must verify the following property (see the previous paragraph): $dist(r_{i+1}, n_i) = dist(r_i, n_i)$. Those two equalities lead to:
$dist(r_0, n_0) = dist(r_0, r_1) + dist(r_1, n_0)$
$dist(r_0, n_0) = dist(r_0, r_1) + dist(r_1, n_1)$
$dist(r_0, n_0) = dist(r_0, r_1) + dist(r_1, r_2) + dist(r_2, n_1)$
$dist(r_0, n_0) = dist(r_0, r_1) + dist(r_1, r_2) + dist(r_2, n_2) = \ldots = \sum_{i=0}^{k-1}(dist(r_i, r_{i+1})) + dist(r_k, n_0)$. (a)
r_0 deflects the traffic sent by r_k to n_k. Thus, $dist(r_k, n_0) = dist(r_k, r_0) + dist(r_0, n_k)$. Furthermore, r_0 prefers the route announced by n_0 to the route announced by n_k, so $\exists K \geq 0, dist(r_0, n_k) = K + dist(r_0, n_0)$ (b) . (a) and (b) lead to:
$\sum_{i=0}^{k}(dist(r_i, r_{i+1})) + K = 0$
Each term of this sum is a positive value, hence:
$\forall i \in \{0, \ldots, k\}, dist(r_i, r_{i+1}) = 0$
If no zero IGP cost circuit exists and if no static route is defined elsewhere on links between an ASBR and its BGP nexthop, the network is loop-free.

Fm-optimality is a bit strong but very helpful to validate a BGP topology. As we will see in section III-B, we can easily compute a lower bound of the number of *fm-optimal* (next-hop,router) pair. If this lower bound is the number of (next-hop,router) pairs, the BGP topology is *fm-optimal* and then all the properties we have just detailed hold. Otherwise, one has to check for each suboptimal pair if problems may occur.

B. Shadowing-reflectors bounding sets

Computing the set $\mathcal{S}(n, r)$ for a pair (n, r) is difficult because BGP route propagation and selection have to be properly simulated inside the AS. We introduce here a more convenient way to detect potential routing deflections in the AS by bounding the set of shadowing reflectors for any pair (n, r). The advantage of relying on the $\mathcal{S}(n, r)$ bounding set is that the BGP protocol does not have to be simulated.

Grey-reflectors: We call *grey-reflector* for pair (n, r) each router r' verifying the following property:

$$\exists n' \in \mathcal{N}, dist(r', n') \leq dist(r', n).$$

A grey reflector may not always advertise the route with next-hop n because it has another *fm-next-hop* that might be preferred if another route is learned. An example of grey-reflector is reported in Figure 8. A shadowing-reflector r' is necessarily a grey-reflector, but this condition is not sufficient. Indeed, in the topology of Figure 9, rr_2 is a grey-reflector for the pair (n, r_3) (because $dist(rr_2, n') \leq dist(rr_2, n)$) but rr_2 cannot learn the route announced by its *fm-next-hop* n'. rr_2 will only learn the route with next-hop n (selected by r_1 and rr_1 and advertises it to r_3. We denote by $\mathcal{G}(n, r) \subseteq \mathcal{R} \backslash \{r\}$ the set of grey-reflectors for any $(n, r) \in \mathcal{N} \times \mathcal{R}$.

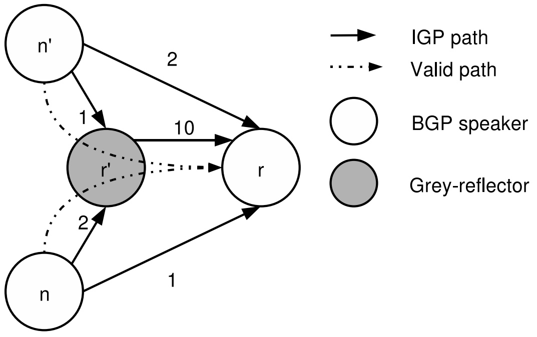

Fig. 8. r' is a grey-reflector for the pair (n, r) because r' has a $n' \neq n$ closest next-hop. If r learns the route with next-hop n', it will not advertise the route with next-hop n.

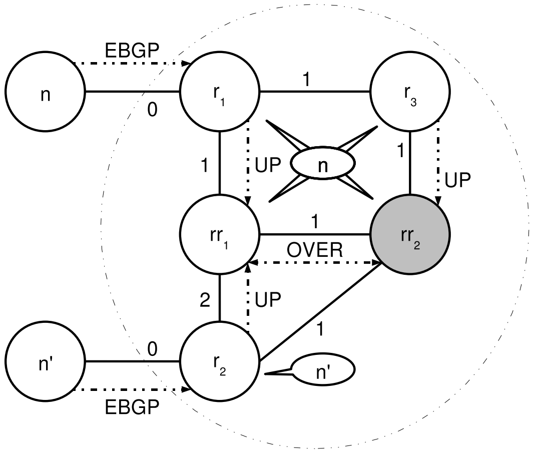

Fig. 9. rr_2 is a grey-reflector for the pair (n, r_3). It advertises the route announced by n because it cannot learn the route announced by n'. rr_2 is hence not a shadowing-reflector for the pair (n, r_3).

Black-reflector: Among routers of the set $\mathcal{G}(n, r)$ with $(n, r) \in \mathcal{N} \times \mathcal{R} \backslash \{r\}$, we can easily detect some shadowing-reflectors. Any border router $r' \in \mathcal{G}(n, r)$ that learns a route with a next-hop $n' \in \mathcal{N}$ by eBGP, will always select the route with n' due to the "eBGP over iBGP" rule. Such border routers are called *black-reflectors* for the (n, r) pair and verify the following property:

$$\exists n' \in \mathcal{N}, n' \neq n \left\{ \begin{array}{l} dist(r', n') \leq dist(r', n) \\ dist(r, n') \geq dist(r, n) \\ \overrightarrow{(n', r')} \in E_{bgp} \end{array} \right.$$

We denote by $\mathcal{B}(n, r) \subseteq \mathcal{R} \backslash \{r\}$ the set of black-reflectors for any $(n, r) \in \mathcal{N} \times \mathcal{R}$ pair.

Shadowing-reflector sets boundaries: Shadowing-reflectors are necessarily grey-reflectors. Black-reflectors are always shadowing-reflectors (and thus a subset of grey-reflectors) because they are always able to learn a route from a concurrent next-hop that has a BGP session with it:

$$\forall (n, r) \in \mathcal{N} \times \mathcal{R}, \mathcal{B}(n, r) \subseteq \mathcal{S}(n, r) \subseteq \mathcal{G}(n, r) \subseteq \mathcal{R} \backslash \{r\}.$$

C. Algorithm to bound *fm-optimality*

Let us choose a router r and one of its *fm-next-hops* n. If r is connected to n in G_{bgp}, *fm-optimality* holds. If not, we compute set of black-reflectors and grey-reflectors for this pair (n, r) as explained in the last section. We remove grey-reflectors from G_{bgp} to obtain $G_{bgp,no-grey}$. We remove black-reflectors from G_{bgp} to obtain $G_{bgp,no-black}$. We build the extended graphs $G^{ext}_{bgp,no-grey}$ and $G^{ext}_{bgp,no-black}$ from the input graphs $G_{bgp,no-grey}$ and $G_{bgp,no-black}$. If a path exists from n to r in $G^{ext}_{bgp,no-grey}$, the pair is necessarily *fm-optimal* because any shadowing-reflector is a grey-reflector. If a path from node n to node r exists in $G_{bgp,no-black}$, the pair may be *fm-optimal*, but not necessarily. We introduce two notations LB and UB such that:

- $LB(n, r) = 1$**:** there exists a path from n to r in $G^{ext}_{bgp,no-grey}$.
- $UB(n, r) = 1$**:** there exists a path from n to r in $G^{ext}_{bgp,no-black}$.

The following algorithm summarizes the steps to compute LB and UB for any pair (n, r). It uses as input a set of *concurrent next-hops* $\mathcal{CN}$. The candidate next-hop sets $\mathcal{CN}$ considered in the algorithm can be reduced to any subset. A set of candidate next-hops used for input in our algorithm can be shared by many prefixes in the real network.

```
1) Input: graphs $G_{bgp}$ and $G_{igp}$, a pair $(n, r)$, a set of next-hops $\mathcal{CN} \subseteq \mathcal{N}$.
2) Output: $LB(n, r)$ and $UB(n, r)$.
3) Check consistency of $G_{bgp}$ (see II-A.3).
4) Build $G^{ext}_{bgp}$ (see II-B).
5)   a) $\mathcal{N}' = \{n' \in \mathcal{CN} \mid dist(r, n') \geq dist(r, n)\}$.
     b) Compute $\mathcal{B}(n, r)$ and $\mathcal{G}(n, r)$ (see III-B).
     c) Compute $G_{bgp,no-black}$ from $G_{bgp}$ and $\mathcal{B}(n, r)$.
     d) Compute $G^{ext}_{bgp,no_black}$ from $G_{bgp,no-black}$.
     e) $UB(n, r) := \begin{cases} 1 \text{ if a path exists from n to r} \\ \quad \text{in } G^{ext}_{bgp,no-black}, \\ 0 \text{ otherwise} \end{cases}$
     f) Compute $G_{bgp,no-grey}$ from $G_{bgp}$ and $\mathcal{G}(n, r)$.
     g) Compute $G^{ext}_{bgp,no-grey}$ from $G_{bgp,no-grey}$.
     h) $LB(n, r) := \begin{cases} 1 \text{ if a path exists from n to r} \\ \quad \text{in } G^{ext}_{bgp,no-grey}, \\ 0 \text{ otherwise} \end{cases}$
```

Note that to simulate IGP failures, we only have to modify the G_{igp} input graph. The first step of the algorithm will remove inconsistent BGP sessions due to the failures.

IV. Experiments on a tiers-1 AS

A. Input data

We study in this section *fm-optimality* of routers to next-hops for the network topology of a large tier-1 AS extracted in March, 2005. We first explain how we build the input topology. Then we explain which candidate next-hop sets we chose to simulate with our algorithm.

IGP topology: The IGP graph model has been built by monitoring the IGP. The IGP graph is a fixed input of our algorithm. This graph can be modified before running our algorithm in order to simulate some internal network failures (when an IGP link is down) and maintenance operations (when a router is down).

BGP topology: We collected the BGP topology from configuration files of each BGP speaker in the AS. We converted these files into the homogeneous C-BGP format [16] and we parsed those C-BGP files to build our reflection graph model. Inconsistent BGP sessions due to the absence of an IGP path or due to a mistake in router configurations are removed from this graph at the third step of the algorithm (see also II-A.3).

B. Results

The algorithm reported in III-C computes $LB(r, n)$ and $UB(n, r)$ for a router r, a next-hop n and a set of candidate next-hops $\mathcal{CN}$. Due to space limitations, we do not show simulation results for IGP failures.

We did not simulate all possible subsets of the concurrent next-hops. Instead, we computed for each router r the set of next-hops that are concurrent with each next-hop n, i.e. for each considered pair (n, r), all next-hops n' that have a larger IGP cost $dist(n', r) \geq dist(n, r)$ are concurrent. This gives us $2, 204$ different sets of concurrent next-hops.

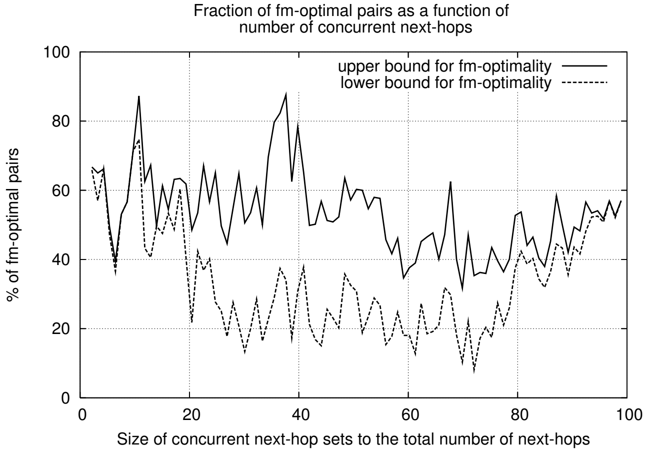

Fig. 10. *fm-optimality* as a function of the number of concurrent next-hops.

We display our simulation results in two different ways. The first way takes the *fm-optimality* computed for each (router, next-hop) pair to the simulated next-hops, and averages the values of LB and UB over all routers of the topology. Figure 10 shows the results in terms of how much the number of considered concurrent next-hops affects the *fm-optimality* over the whole AS. Figure 10 plots the fraction of *fm-optimal* (router, next-hop) pairs as a function of the set size of concurrent next-hops. We see in Figure 10 that even for set sizes of less than 10 concurrent next-hops, a significant fraction of potential *fm-suboptimality* exists in the AS, between 30 and 60%. We also see on most set sizes that there can be large variations between worst-case and best-case *fm-optimality*. This indicates that different routers and next-hop pairs may exhibit different *fm-optimality* behaviors.

The second way to display our results is to average the *fm-optimality* computed for each router and next-hop pair to the simulated next-hops, over the different next-hops of the topology. Figure 11 shows the routers, ordered by decreasing average *fm-optimality*. We see on Figure 11 that even the routers having the highest *fm-optimality* exhibit a significant percentage of *fm-suboptimal* pairs, at least 30%. All routers have a lot of potential *fm-suboptimality*, between 30% and 50% of suboptimal pairs in the best-case.

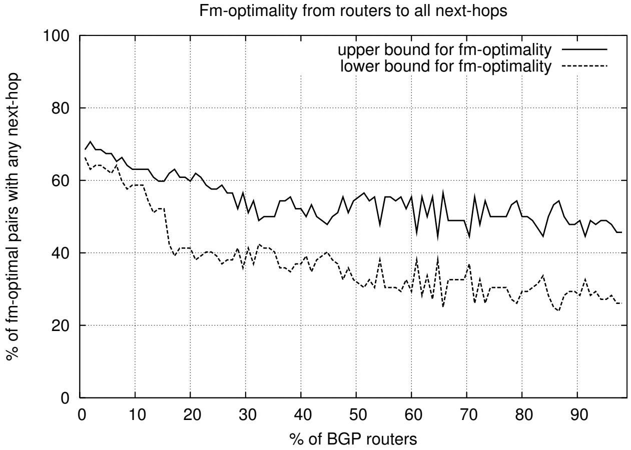

Fig. 11. *fm-optimality* per router.

The previous two figures provided results in terms of all possible concurrent next-hops. Due to routing policies implemented by ASs, concurrent routes are typically those received from a single neighbor that is preferred to all others. In such a case, the number of concurrent next-hops to be considered is restricted to those of a single neighboring AS. We show on Figure 12 the fraction of *fm-optimal* (router, next-hop) pairs towards next-hop sets restricted to a single neighboring AS, for each router (ordered by decreasing average fraction of *fm-optimality*). We observe that the *fm-optimality* is in such cases better than when considering larger sets of concurrent next-hops. This indicates that the *fm-optimality* that BGP might undergo is probably better than what an arbitrary set of concurrent next-hops would give.

V. Conclusion

We proposed in this paper to check for the optimality of the routes choice in a route reflection graph. We do so without requiring to simulate the complex operation of the BGP protocol. We formalized the problem of detecting potential suboptimal choices of the BGP routes. We introduced the concept of *shadowing reflector*, routers that are the cause of suboptimal route choices in an AS. We show that checking for the existence of *shadowing reflectors* is tricky, and in some cases requires to simulate the BGP protocol.

We then applied our check procedure to a large tier-1 AS. We showed that up to 30% of potentially suboptimal routes can happen with realistic sets of concurrent next-hops.

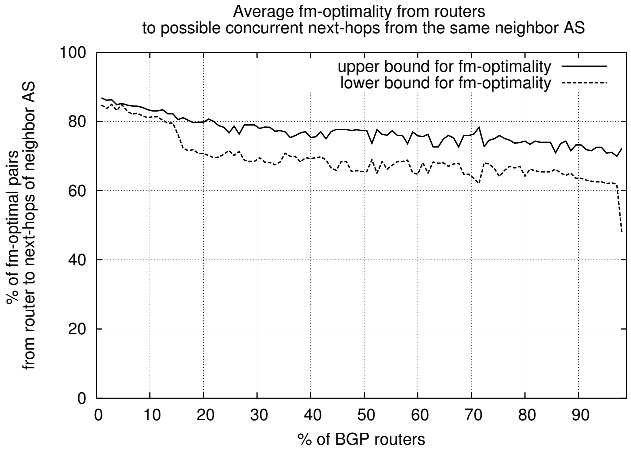

Fig. 12. *fm-optimality* per router when the set of next-hops is restricted to a single neighboring AS.

Checking for optimal points in iBGP is actually part of a larger problem: designing iBGP topologies that will automatically prevent suboptimal routing, deflection and loops from occurring, especially in the case of failures. As further work, we plan to investigate the different ways of designing such iBGP topologies.

References

[1] T. G. Griffin and G. Wilfong, "On the correctness of iBGP configuration," in *Proc. of ACM SIGCOMM*, August 2002.

[2] N. Feamster and H. Balakrishnan, "Detecting BGP Configuration Faults with Static Analysis," in *Proceedings of the 2nd Symposium on Networked Systems Design and Implementation (NSDI)*, May 2005.

[3] T. Bates, R. Chandra, and E. Chen, "BGP Route Reflection - An Alternative to Full Mesh IBGP," Internet Engineering Task Force, RFC2796, April 2000.

[4] B. Halabi and D. M. Pherson, *Internet Routing Architectures (2nd Edition)*. Cisco Press, January 2000.

[5] R. Teixeira, T. Griffin, G. Voelker, and A. Shaikh, "Network sensitivity to hot potato disruptions," in *Proc. of ACM SIGCOMM*, August 2004.

[6] Cisco, "BGP best path selection algorithm," http://www.cisco.com/warp/public/459/25.shtml.

[7] S. Uhlig and S. Tandel, "Quantifying the impact of route-reflection on BGP routes diversity inside a tier-1 network," in *Proc. of IFIP Networking*, Coimbra, Portugal, May 2006.

[8] N. Feamster, J. Winick, and J. Rexford, "A Model of BGP Routing for Network Engineering," in *ACM Sigmetrics - Performance 2004*, New York, NY, June 2004.

[9] A. Feldman, H. Kong, O. Maennel, and A. Tudor, "Measuring BGP pass-through times," in *PAM*, 2004, pp. 267–277.

[10] A. Rawat and M. A. Shayman, "Preventing persistent oscillations and loops in iBGP configuration with route reflection," *Computer Networks*, pp. 3642–3665, December 2006.

[11] M. Vutukuru, P. Valiant, S. Kopparty, and H. Balakrishnan, "How to construct a correct and scalable iBGP configuration," in *IEEE INFOCOM*, Barcelona, Spain, April 2006.

[12] L. Xiao, J. Wang, and K. Nahrstedt, "Optimizing iBGP route reflection network," in *IEEE INFOCOM*, 2003.

[13] T. Griffin and G. Wilfong, "On the Correctness of iBGP Configuration," in *Proceedings of SIGCOMM*, August 2002.

[14] N. Feamster, J. Winick, and J. Rexford, "A model of BGP routing for network engineering," in *Proc. of ACM SIGMETRICS*, June 2004.

[15] M. Buob, M. Meulle, and J. Lutton, "Un modèle de graphe et de dioide pour le routage interdomaine," *CFIP 06*, October 2006.

[16] B. Quoitin and S. Uhlig, "Modeling the Routing of an Autonomous System with C-BGP," *IEEE Network Magazine*, vol. 19, no. 6, November 2005.