2016 IEEE 41st Conference on Local Computer Networks

Achieving Stable iBGP with Only One Add-Path

Xiaomei Sun*‡, Qi Li†‡, Mingwei Xu*‡ and Yuan Yang*‡
*Dept. of Computer Science and Technology, Tsinghua University, China
†Graduate School at Shenzhen, Tsinghua University, China
‡Tsinghua National Laboratory for Information Science and Technology
sunxm15@mails.tsinghua.edu.cn qi.li@sz.tsinghua.edu.cn xumw@tsinghua.edu.cn yyang@csnet1.cs.tsinghua.edu.cn

***Abstract*—Border Gateway Protocol (BGP) has been and will still be the de-facto standard for inter-domain routing in the Internet. However, the problem of routing oscillations in BGP has not been well addressed, which can introduce lots of unnecessary routing updates and severely degrade network performance. In particular, existing studies need a great effort to be deployed or introduce a large overhead. In this paper, we propose to first detect a routing oscillation quickly after the oscillation happened, and then, we eliminate the routing oscillation by disseminating only one additional path (Add-path). Based on analysis of BGP updates in the routers where oscillations have already happened, we present a general method to detect a routing oscillation within a couple of routing replacements. Then, we show that one more Add-path is enough to stop the oscillation. We propose the Minimal Add-paths BGP (MA-BGP) approach, develop algorithms, and prove that MA-BGP can guarantee stable iBGP by a classical model that captures the underlying semantics of any path vector protocol including BGP. The simulation results show the effectiveness and efficiency of our approach.**

***Keywords*—iBGP; oscillations; one Add-path; stability**

I. Introduction

Border Gateway Protocol (BGP) [1] has been and will still be the de-facto standard for inter-domain routing in the Internet. Nevertheless, it is well accepted that BGP is not robust enough against routing dynamics. In particular, routing oscillations may occur in BGP, which prevent the routing tables of certain routers from converging to a stable solution, and thus degrade the performance of packet forwarding, or even damage the reachability [2–6]. Routing oscillations can also trigger a large number of BGP update messages, which consume the network resources such as CPU, memory, and bandwidth unnecessarily.

There are two sub-protocols of BGP. External Border Gateway Protocol (eBGP) is used to share routing information between neighboring BGP routers that belong to different Autonomous Systems (AS-es), while internal Border Gateway Protocol (iBGP) is used to exchange external routing information among routers within the same AS. People have found that BGP routing oscillations are caused by certain features within iBGP [7], such as Multi-Exit Discriminator (MED) and Route Reflection Clustering [8]. Route reflection introduces routing information hiding and decreases path diversity, while MED with routing reflection may lead to a non-transitive ordering of routes and aggravate routing anomalies.

Existing studies tried to enable stable iBGP by preventing routing oscillations from happening from the very beginning. For instance, Griffin et al. [3], Vutukuru et al. [9] and Buob et al. [10] proposed to configure iBGP correctly with better parameters. These approaches addressed this issue by constructing correct, scalable configurations or designing optimal route-reflection topologies. However, it would need a great effort to modify configurations to adopt the approaches because of the deployment costs. Flavel et al. [7] proposed to modify the route decision process of iBGP. Basu et al. [4] proposed that the route reflectors retransmit all routes instead of the best one. These approaches introduce a large overhead, because routing oscillations may occur in many different situations, and it is difficult to cover them all at once.

To address this issue and achieve stable iBGP with little overhead, we take a different approach in this paper. That is, we first detect a routing oscillation quickly after the oscillation happened, and then, we eliminate the routing oscillation in a more targeted way, by disseminating an additional path (Add-path). There are two key findings to make this possible. First, we find a general method to detect a routing oscillation within a couple of routing replacements. This is based on analysis of BGP update pattern in the routers where oscillations have already happened, and we find that our detection condition is general for different types of oscillations. Second, we find that as soon as the routing oscillation is known, one more Add-path is enough to stop the oscillation. Thus, much less overhead will be introduced comparing with advertising all possible paths [4].

We propose Minimal Add-paths BGP (MA-BGP), a lightweight approach to achieve stable iBGP. MA-BGP can guarantee iBGP stability by disseminating only one Add-path only when necessary and does not change iBGP policies and messages types. We show the necessary and sufficient condition of iBGP routing oscillations, based on which we develop an algorithm to detect a routing oscillation and select an Add-path to disseminate. We demonstrate that MA-BGP can guarantee stability by the classical Dispute Wheel [5] model which captures the underlying semantics of any path vector protocol including BGP. We also analyze the overhead. We implement MA-BGP in C-BGP [11] and evaluate the performance by simulations with synthetic and real topologies. The results show that MA-BGP is able to eliminate iBGP routing oscillations. The total number of routes propagated in the network is reduced by more than 90%; besides, no other routing problems are introduced but a slight overhead for the topologies without oscillations.

DOI 10.1109/LCN.2016.119

TABLE I: Decision Process

Step	Attribute
1	Highest Local-Preference
2	Shortest AS-Path
3	Lowest ORIGIN
4	Lowest MED
5	eBGP over iBGP
6	Nearest IGP distance
7	Lowest ROUTER-ID
8	Shortest CLUSTER-ID-LIST
9	Lowest neighbor address

The rest of the paper is organized as follows. Section II gives a background of iBGP routing oscillations and reviews related work. A necessary and sufficient condition of iBGP oscillations is presented in Section III. Section IV is dedicated to the development of MA-BGP, followed by the correctness and overhead proofs of MA-BGP in Section V. Section VI shows the methods and results of the performance evaluations, and Section VII concludes the paper.

II. Background and Related work

In this section, we introduce two well known cases of iBGP routing oscillations in the network, including MED oscillations and topology oscillations, and then we briefly review the previous solutions to this problem.

A. Overview of iBGP

We begin with a brief overview of the iBGP protocol and predefine some notations used in the rest of the paper.

iBGP Overview. iBGP is used to exchange external routing information among routers within the same AS. Each iBGP router selects its best route by the decision process. To prevent looping in the announcement, iBGP maintains a full mesh of sessions which does not scale well. An alternative solution to this problem is called Route Reflection Clustering [8].

Decision Process. The decision process is to select an optimal route among all the routes in Adj-RIB-In (Adjacent Routing Information Base, Incoming). If the selected best route is different from the one before this decision process, the router will update its Loc-RIB (Local Routing Information Base) and the Adj-RIB-Out (Adjacent Routing Information Base, Outgoing) for each neighbor. Then, the router will disseminate the corresponding route to all the neighbors. The decision process steps is presented in TABLE 1 [12]. BGP speakers select the optimal path following these steps. Among all the attributes, MED value is nontransitive, which would obtain different preferences for the two same paths when they belong to different paths sets, thus causing BGP vulnerable to routing oscillations [6]. The non-transitivity of MED only affects the subsequent attributes in the decision process.

Route Reflection Clustering. The main idea of Route Reflection Clustering is to use a two-level hierarchy. The routers in an AS are divided into a collection of disjoint sets called *clusters*. Each cluster consists of one or more special routers called *route reflectors* and all other routers in the cluster are *clients* of the reflectors.

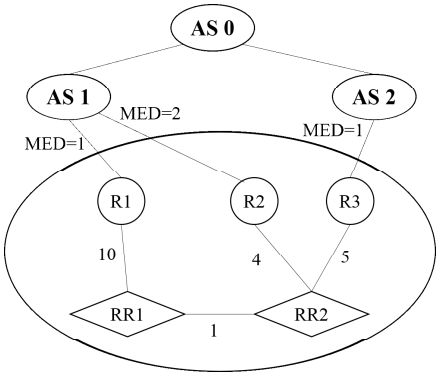


Fig. 1: MED oscillation with route reflection. In this example, IGP topology and iBGP topology overlap completely, so the dashed lines are omitted.

Notations. A *path* P_{R_i} (or a route) is a sequence of nodes $< R_i, R_j, ... >$, where R_i is the next hop. RR represents router reflector while R denotes clients. $P_{R_i} \in RR_i$ means that P_{R_i} belongs to the Adj_RIB_In of RR_i and $P_{R_i} > P_{R_j}$ means that P_{R_i} is preferred over P_{R_j}.

B. iBGP Oscillations

MED Oscillation. BGP uses MED to differentiate multiple links connecting the same pair of AS-es. However, as has been observed [13], the key problem in persistent route oscillation (under route reflection scenarios) is just the use of MED attribute for route comparison. Since MED values are not used to compare routes that pass through different neighboring AS-es, the use of MED values may periodically hide certain routes from view and lead to non-transitive routes comparison. In combination with route reflection, route oscillations are likely to happen.

A typical example of MED oscillation is as follows in Fig. 1 [14]. In the figures of this paper, AS-es are shown by eclipses, where the biggest one represents the AS in which we analyze how iBGP works. Route reflectors are shown by diamonds and clients by circles. Solid lines are IGP links and the numbers on the lines are IGP distances. Dashed lines are iBGP sessions. In Fig. 1, oscillations arise in the two reflectors RR_1 and RR_2, where the preference lists are

$$P_{R_3} > P_{R_1} > P_{R_2} \tag{1}$$

and

$$\begin{cases} P_{R_2} > P_{R_3}, \text{ if } P_{R_2}, P_{R_3} \in {}^{1}RR_2 \\ P_{R_3} > P_{R_1} > P_{R_2}, \text{ if } P_{R_1}, P_{R_2}, P_{R_3} \in RR_2 \end{cases} \tag{2}$$

respectively. The preference between P_{R_2} and P_{R_3} in $RR2$ is inconsistent, which is caused by MED's non-transitivity. We start from that clients' routes are preferred, that is, RR_1 chooses route P_{R_1} from R_1 and RR_2 chooses route P_{R_2} from R_2 respectively as their optimal routes, then they exchange

[1]Here, $P \in R$ means that router R has learned the route P.

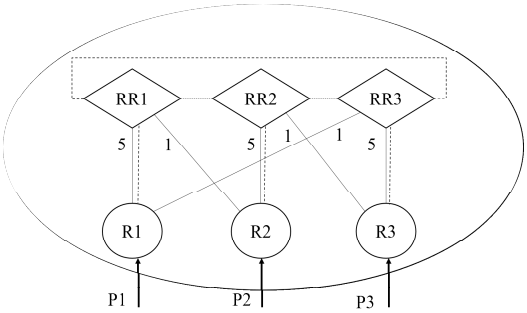


Fig. 2: iBGP topology oscillation. The arrows in the bottom (P1,P2,P3) are eBGP routes from the same source AS.

their best routes information. When RR_2 learns route P_{R_1}, RR_2 changes its best route to P_{R_3} and advertises it to RR_1, because P_{R_1} eliminates P_{R_2} when comparing MED values and P_{R_3} eliminates P_{R_1} when comparing IGP distances. After RR_1 learns P_{R_3}, it chooses P_{R_3} as best on account of smaller IGP distance and withdraws P_{R_1} to RR_2. Then RR_2 decides P_{R_2} to be best again because P_{R_1} is no longer available. And RR_1 changes its best to P_{R_1} again after RR_2 advertises P_{R_2} again. Now, we have come back to the beginning and get into a persistent cycle of oscillation which never converges to a stable routing solution.

Topology Oscillation. Topology oscillation in iBGP is caused by the interaction between the logical iBGP topology and the physical IGP topology, which are usually not corresponding. iBGP configuration determines how routes propagate while the route selection process is based on the IGP distance. Consequently, the inconsistency may lead to circular dependencies which may oscillate [7].

We give a simple example in Fig. 2 [14], where each reflector prefers a route from another cluster than that from its client. Starting within clusters, RR_i chooses the route P_{R_i} from its client as best and disseminates to other reflectors. When reflector RR_i learns a route $P_{R_{(i+1)\%3}}$ from $RR_{(i+1)\%3}$, RR_i selects the new route $P_{R_{(i+1)\%3}}$ as optimal and withdraw its own client's route P_{R_i}. Then, all reflectors choose a route from another cluster as best and no longer advertise routes from their own clients. Thus, RR_i only knows routes within its cluster and changes its best route to P_{R_i}. Now we have gone back to the beginning and will get into a persistent cycle of oscillation forever.

C. Related work

iBGP has also been an area of much investigation. Various solutions [3, 4, 7, 9, 10] have been proposed to address the iBGP convergence problem. Griffin et al., Vutukuru et al. and Buob et al. proposed to configure iBGP correctly. Griffin et al. gave simple sufficient conditions on network configurations that guarantee correctness. Vutukuru et al. focused on the construction of an iBGP session configuration that guarantees two correctness properties - loop-free forwarding paths and complete visibility to all eBGP-learned best routes. Buob et al. came up with a solution to design iBGP route-reflection topologies which lead to the same routing as with an iBGP full-mesh. Besides, Flavel et al. proposed to modify the route decision process of iBGP. Basu et al. proposed that the route reflectors retransmit all routes instead of the best one.

A critical reason of routing oscillation in iBGP is the poor path diversity at the BGP router level, as described in [15]. Therefore, Schrieck et al. [15] proposed a way named Add-Paths to solve this issue by advertising multiple paths over iBGP sessions. They analyzed the various options for the selection mode of the paths to be advertised and found that these modes differently fulfill the needs of Add-Paths applications including oscillations avoidance. The Advertise All Paths Mode can prevent routing oscillations with expensive storage and transmission cost. The Advertise N Paths Mode helps to reduce routing oscillations, but not in all cases. The Advertise All AS-Wide Best Paths Mode prevents MED oscillations which is similar with [7].

Some of the above solutions require a lot of changes to BGP protocol and others introduce expensive overhead for every router in the network. By detecting oscillations and adding little change to BGP dissemination process, we come up with a new dynamic way in this paper: advertising one more path merely on the specific nodes wherever an oscillation is detected. In this way, we can say that minimal communication overhead is introduced, that is, one extra path for one oscillation in a node.

III. Oscillation Detection Condition

In this section, we summarize a necessary and sufficient detection condition by analyzing the general pattern of the route updates when iBGP oscillations occur, which can be used to detect both MED and topology oscillations in iBGP with route reflection.

A persistent routing oscillation is thought to happen when the following two procedures are infinitely alternately satisfied. If they only appear several times and then settle to a stable routing table, then we call it a transient oscillation, which may be caused by timing coincidence [4].

- Procedure a. Sub-optimal route replaces the optimal route.
  Here is the concrete scenario where we call the sub-optimal route replaces the optimal one. Router R receives updates from its neighbors. After the decision process, a new best route is selected in R. Nevertheless, regardless of other routes in R, when we compare the new best route directly with the old one using the decision process described in TABLE 1, the latter turns out to be better. The reason of this kind of replacement usually is MED non-transitivity or route withdrawing. In Fig. 1, there is a sample replacement because of nontransitive MED. P_{R_3} replaces P_{R_2} as the new optimal route in RR_2, but P_{R_2} is actually better than P_{R_3} because of smaller IGP weight.
- Procedure b. A replaced optimal route is selected as the best again in the router.
  After a decision process in one router, the selected best route has ever been optimal once. For example in Fig. 1,

P_{R_2}, which was replaced by P_{R_3} before, now becomes the optimal path again after RR_1 withdraws P_{R_1}.

Then, we can obtain the following theorem stating the necessary and sufficient condition for iBGP routing oscillations.

Theorem 1.(Oscillation Detection Condition) An iBGP routing oscillation occurs if and only if the above two procedures alternate constantly in one router.

Proof: Now, we prove the sufficiency and necessity of the condition as follows.
Sufficiency: If the above condition is satisfied, then an oscillation happens.
The sufficiency of the detection condition is obvious. If the best route is alternately replaced by another and selected as optimal again and again, by definition, it is an oscillation that happens.
Necessity: If there is an oscillation in the network, then the above condition will be satisfied.
The oscillations can be divided into MED oscillations and topology oscillations, which will be analyzed respectively.

- If it is a MED oscillation, there must be three routes $P_{R_1}, P_{R_2}, P_{R_3}$ in a router R, which satisfies that (1) P_{R_1}, P_{R_2} are from distinct AS-es. P_{R_2}, P_{R_3} are from the same AS. (2) $P_{R_3} <$ [2]P_{R_1} and $P_{R_1} < P_{R_2}$ because of smaller IGP distance, or other attributes after MED. (3) $P_{R_2} < P_{R_3}$ because P_{R_3} has lower MED. (4) Only P_{R_3} is advertised by another cluster. Thus, the best route will change between P_{R_1} and P_{R_2}. Procedure a is satisfied when P_{R_1} replaces P_{R_2} and procedure b when it is the other way around.
- If it is a topology oscillation, that is, there would be a circular set of reflectors where each one prefers a route from another cluster rather than from its own client. Suppose RR_i is one of the reflectors, then there is P_{R_i} and $P_{R_{i+1}}$ representing the routes from its client and reflector RR_{i+1} respectively. $P_{R_i} < P_{R_{i+1}}$ but $P_{R_{i+1}}$ will always be withdrawn when it is chosen by RR_i. Hence, procedure a is met when $P_{R_{i+1}}$ is withdrawn and procedure b when $P_{R_{i+1}}$ is received again.

In conclusion, the condition will be satisfied when an oscillation occurs in the network. □

IV. The MA-BGP Approach

In this section, we propose a new approach named MA-BGP based on Add-paths to solve routing oscillation problems in iBGP.

A. Basic Idea

For the convenience of explanation, we introduce some important definitions in MA-BGP.

[2]In this section, the less-than sign means lower prioity when comparing both routes using decision process.

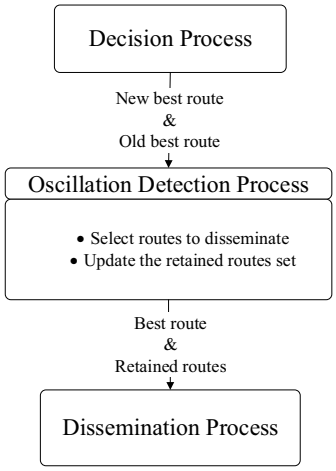


Fig. 3: Modified BGP module: an oscillation detection process is added between decision process and route dissemination process.

Retained routes Set. Every router has a retained routes set, keeping all the routes related to oscillations.

Oscillation detected. When an optimal route is replaced by a sub-optimal one, we call an oscillation has been detected.

Oscillation occurred. When a replaced route is selected as the best again, we call an oscillation has occurred in this router.

According to the detection condition, an oscillation in a configuration is that two routers keep exchanging their best routes again and again. The result of one router's decision process is influenced by another router's result and vice versa. Hence in a single router, it behaves as infinitely changing its optimal route from one to another alternately. Thereupon, we come up with the following idea.

When the best route is updated after the router's decision process, we add *an oscillation detection process* before the routes dissemination process. Fig. 3 depicts the modified BGP module.

In BGP-4 [1], there is at most one route for each neighbor in Adj-RIB-In and Adj-RIB-Out, which has turned out to be vulnerable to oscillations. In MA-BGP, multiple routes are allowed to be disseminated to the neighbors. When the best route of a given router is updated after its decision process, we detect oscillations before the dissemination process. The result of the detection process can be divided into three types. (1) An oscillation is $detected$. This routing update situation reaches detection condition and thus records the old best route in retained routes in case that it would be selected as the optimal again next time. (2) An oscillation $occurs$. A route in the router's retained routes has become the best route in this decision process, we add the old best route into the retained routes set and disseminate the set with the best route simultaneously. (3) $Normal$ update. It is only a new better route replaces the old one. The new best route would be disseminated as original BGP and nothing needs to be done to the retained routes now.

In this way, we propagate a pair of routes for every

oscillation and thus stop all kinds of iBGP routing oscillations.

B. Algorithm

ALGORITHM 1 MA-BGP

Input: $newBest$, $oldBest$, $retainedRoutes$
1: **function** DETECTION_PROCESS($newBest, oldBest, retainedRoutes$)
2: $wait\_update(newBest, oldBest)$
3: $state \leftarrow osci\_detection(newBest, oldBest, retainedRoutes)$
4: **switch** $state$ **do**
5: **case** $DETECTED$
6: $disseminate(newBest, retainedRoutes)$
7: $retainedRoutes.append(oldBest)$
8: **case** $OCCUR$
9: $retainedRoutes.append(oldBest)$
10: $disseminate(newBest, retainedRoutes)$
11: **case** $NORMAL$
12: $disseminate(newBest, retainedRoutes)$
13: **go to** 2
14: **end function**

ALGORITHM 2 Oscillation Detection

1: **function** OSCI_DETECTION($newBest, oldBest, retainedRoutes$)
2: $updateIsNormal \leftarrow compare\_routes(newBest, oldBest)$
3: **if** $newBest.LocalPref = oldBest.LocalPref$ **and** $newBest.ASPath = oldBest.ASPath$ **and** $updateIsNormal = False$ **then**
4: **return** $DETECTED$
5: **else if** $newBest \in retainedRoutes$ **then**
6: **return** $OCCUR$
7: **else**
8: **return** $NORMAL$
9: **end if**
10: **end function**

ALGORITHM 3 Compare two routes

1: **function** COMPARE_ROUTES($newBest, oldBest$)
2: **if** (MED $unused$ **or** $incomparable$ **or** $equal$) **and** (($newBest$ is $eBGP$ **and** $oldBest$ is $iBGP$) **or** ($equal$ $before$ $igpCost$ **and** $newBest.igpCost < oldBest.igpCost$) **or** ($equal$ $before$ $nexthop$ **and** $newBest.nexthop < oldBest.nexthop$) **or** ($equal$ $before$ $clusterlist$ **and** $newBest.clusterlist < oldBest.clusterlist$) **or** ($equal$ $before$ $neighborip$ **and** $newBest.neighborip < oldBest.neighborip$)) **then**
3: **return** $True$ /* $newBest$ is $better$ */
4: **else**
5: **return** $False$ /* $oldBest$ is $better$ */
6: **end if**
7: **end function**

MA-BGP algorithm describes how a router disseminates routes to neighbors, in our modified BGP protocol, when its best route is updated. Firstly, the algorithm waits for a best route update, upon which the $osci\_detection$ function is called to check whether an oscillation is detected or already has occurred. If the result of $osci\_detection$ turns out to be $detected$, the router advertises the $newBest$ and the $retainedRoutes$ (what has been advertised before for other ocsillations, maybe empty if no oscillations have ever happened) simultaneously, and then append the $oldBest$ into $retainedRoutes$ set; if the result is $occurring$, the router adds $oldBest$ into the $retainedRoutes$ and advertises the $newBest$ as well as the updated $retainedRoutes$ set; otherwise, this update is thought to be a normal one. The router only advertises the $newBest$ route to neighbors like the original iBGP along with the $retainedRoutes$, which is recorded before for some other oscillations. No modification is need to be done to $retainedRoutes$ set here.

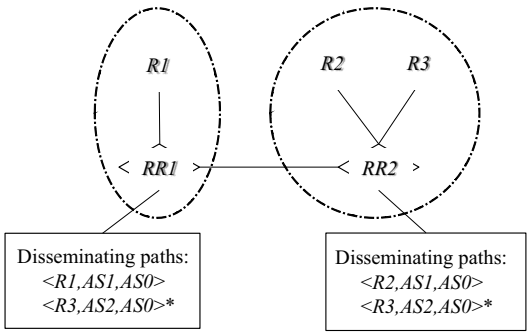


Fig. 4: Key component of MED oscillation. The paths to be disseminated according to MA-BGP are presented, where $*$ means the best one in the router.

C. MA-BGP Applied Cases

In this section, we analyze two corresponding examples to the well-known cases in Section II.

Solving MED Oscillation. As we already know, MEDs' incomparability as well as route-reflector hierarchy's information hiding lead to MED oscillations. For example in Fig. 4, which is the critical component of a MED oscillation, $RR1$ and $RR2$ are adjacent reflectors. R_1, R_2 and R_3 are their clients. If the reflectors exchange routes in the following way, there will be a stable routing solution.

- In router RR_1, route P_{R_3} is selected as optimal, and route P_{R_1} is also disseminated to its neighbors at the same time.
- In router RR_2, route P_{R_3} is selected as optimal, and route P_{R_2} is disseminated to neighbors along with P_{R_3} simultaneously.

Solving Topology Oscillation. The interaction between the route-reflector iBGP topology and the IGP leads to topology oscillation. For example in Fig. 5, which is the critical component of an iBGP topology oscillation, RR_1 and RR_2 are adjacent reflectors and R_1, R_2 are their clients respectively. If the reflectors exchange routes in the following way, there will be a stable routing solution.

- In router RR_1, route P_{R_2} is the best path, simultaneously with route P_{R_1} received from its client disseminated to its neighbors.

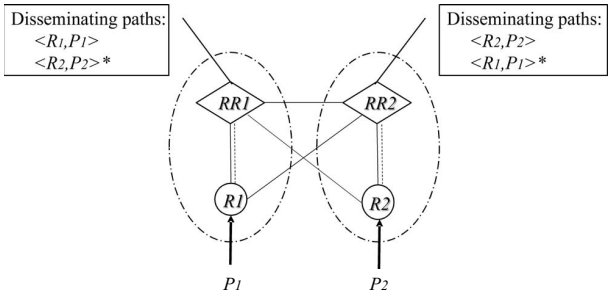


Fig. 5: Key component of iBGP topology oscillation.

- In router RR_2, route P_{R_1} is selected to be the best path, simultaneously with route P_{R_2} received from its client disseminated to its neighbors.

V. Theoretical Analysis

In this section, we analyze our algorithm from two aspects. Firstly, it is proved to be able to get rid of routing oscillations. Furthermore, minimal extra paths are introduced in our algorithm.

A. Stability

The proof that our algorithm could guarantee route stability is based on the Stable Paths Vector Protocol (SPVP), which is a distributed algorithm for solving the Stable Paths Problem (SPP) [5]. According to Griffin et al., SPP captures the underlying semantics of any path vector protocol such as BGP. A solution to the SPP is an assignment of permitted paths to nodes so that each node's assigned path is its highest ranked path extending any of the assigned paths at its neighbors. SPVP can solve SPP in a distributed manner. And Dispute wheel is a derived structure representing a circular set of dependencies between routing policies that cannot be simultaneously satisfied. Griffin et al. concluded the following.

Theorem 2. [5] The lack of dispute wheels guarantees that SPVP could converge to the unique solution of the corresponding SPP, and therefore, guarantees protocol convergence.

Based on the above theorem, we prove that our algorithm can achieve BGP stability. Comparing the two instances of SPP, BAD GADGET and BAD BACKUP [5], we can find that both of them have a dispute wheel, but BAD BACKUP has a unique solution and BAD GADGET has none. For the latter, the SPVP protocol will always diverge.
Thereby, we can easily come to the following definition.

Definition 1. Bad Dispute Wheel
A bad dispute wheel, depicted in Fig. 6 [5], π=($\vec{R}$, $\vec{p}$, $\vec{r}$), of size k, is a sequence of nodes $\vec{R}$ =R_0,R_1,...,R_{k-1}, and sequences of nonempty paths $\vec{p}$= p_0,p_1,...,p_{k-1} and $\vec{r}$=r_0,r_1,...,r_{k-1}, such that for each $0 \leq i \leq k-1$ we have (1) r_i is a path from R_i to R_{i+1}, (2) p_i is a permitted path from node R_i to the origin, (3) $r_i p_{i+1}$ is a permitted path from R_i to the origin, (4) $r_i p_{i+1}$ is preferred than p_i, and *(5)* $r_i p_{i+1}$ *has the highest preference among all permitted paths from* R_i *to the origin.* (All subscripts are to be interpreted modulo k.)

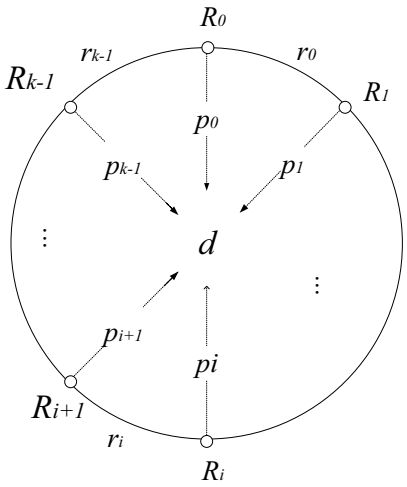


Fig. 6: A bad dispute wheel of size k with the origin d. Path $r_i p_{i+1}$ has the highest priority in node R_i to the origin d.

Lemma 1. No bad dispute wheel implies route convergence.
Proof: According to the proof of Theorem V.9 in [5], we can have exactly the same deduction. Let S be an instance of SPP. Suppose that S diverges, Griffin et al. have shown that S contains a dispute wheel. All we have to do here is to prove the dispute wheel is a bad one. Actually, from their proof, it has been mentioned that path $r_i p_{i+1}$ is of highest rank at R_i, which satisfies condition (5) in Definition 1. □

Theorem 3. A bad dispute wheel produces route oscillations.
Proof: Obviously. It is impossible that, for each node, $r_i p_{i+1}$ is chosen to be the best simultaneously. The best path changes between $r_i p_{i+1}$ and another path continuously, thus producing oscillations. □

Theorem 4. MA-BGP is able to eliminate all bad dispute wheels.
Proof: The oscillations caused by a bad dispute wheel, as Theorem 3, will be detected by the detection condition and trigger our algorithm. At each node $R_{i+1}, 0 \leq i \leq k-1$, $r_{i+1} p_{i+2}$ is chosen as the best path without withdrawing p_{i+1}, which allows node R_i could select $r_i p_{i+1}$ as the best one at the same time. Informally, a circular set of dependencies between routing policies are satisfied simultaneously by adding disseminated paths. Therefore, every bad dispute wheel will be broken as last. □

Theorem 5. MA-BGP guarantees route stability.
Proof: By combining Lemma 1 and Theorem 4, we can draw to this conclusion easily. □

B. Minimal Add-Paths

To solve routing oscillations by using Add-Paths, it is inevitable to bring about network communication overhead, which is the less cost, the better. MA-BGP in this paper, compared with the previous solutions, has strengths in the following two aspects.

- The amount of nodes disseminating multiple paths, denoted by n.
- The amount of extra paths advertising by each node, denoted by p.

Then we can come up with the following theorem.

Theorem 6. MA-BGP introduces the minimal Add-Paths.
Proof: Suppose there is a network of N nodes, where our algorithm is deployed. If an oscillation happens, which means that a bad dispute wheel of size n ($n \leq N$) exists. According to MA-BGP, not all nodes in the network, but only those where oscillations are detected, that is, R_i, $0 \leq i \leq n-1$, in the bad dispute wheel, will trigger Add-Paths behavior. The number of extra paths can be displayed in this way.

$$\begin{cases} p = 1, \ for\ nodes\ where\ an\ oscillation\ occurs. \\ p = 0, \ for\ nodes\ where\ no\ oscillations\ happen. \end{cases} \tag{3}$$

Only $p = 1$ extra path, that is, p_i is appended to the nodes in oscillation, R_i, $0 \leq i \leq n-1$.

From the view of the whole network, there is *less or equal to one path* added to each node in one oscillation. Therefore, our algorithm could be classified as Add $p \leq$1 Path Mode and regarded as a solution of minimal transmission overhead. □

VI. Performance Evaluation

In this section, we introduce how the algorithm MA-BGP is implemented in an Open-source solver for BGP, i.e., C-BGP and then detail our simulation and evaluation on it.

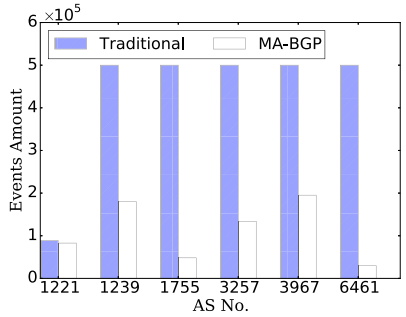
A. Implementation and Simulation Setup

C-BGP is an open source efficient BGP solver developed by Bruno Quoitin written in C, consisting of NET module, BGP module, SIM module, etc. We mainly study the BGP module where the BGP protocol is implemented and then insert our algorithm here.

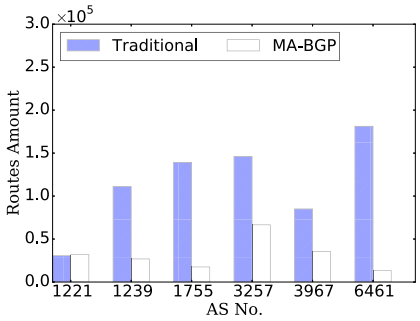
We use two sets of topologies in our simulations.

- Small-scale random topologies.
  In this case, the number of routers usually ranges from 5 to 100. The nodes and IGP physical links between them are assigned by program randomly. And then configure the iBGP logical topology with route reflection according to a regular pattern.[3] We conduct hundreds of small-scale experiments to verify the correctness by checking the routes updating process event by event.

[3]The pattern is designed in this way. First, all the routers are divided into several clusters, each with 2 routers at least. Then, a reflector needs to be selected for every cluster. We compute the sum of the distances from one router to all others in its cluster and pick the one with the minimal sum as the reflector. And the other routers are clients of the reflector.



(a) Events Amount



(b) Routes Amount

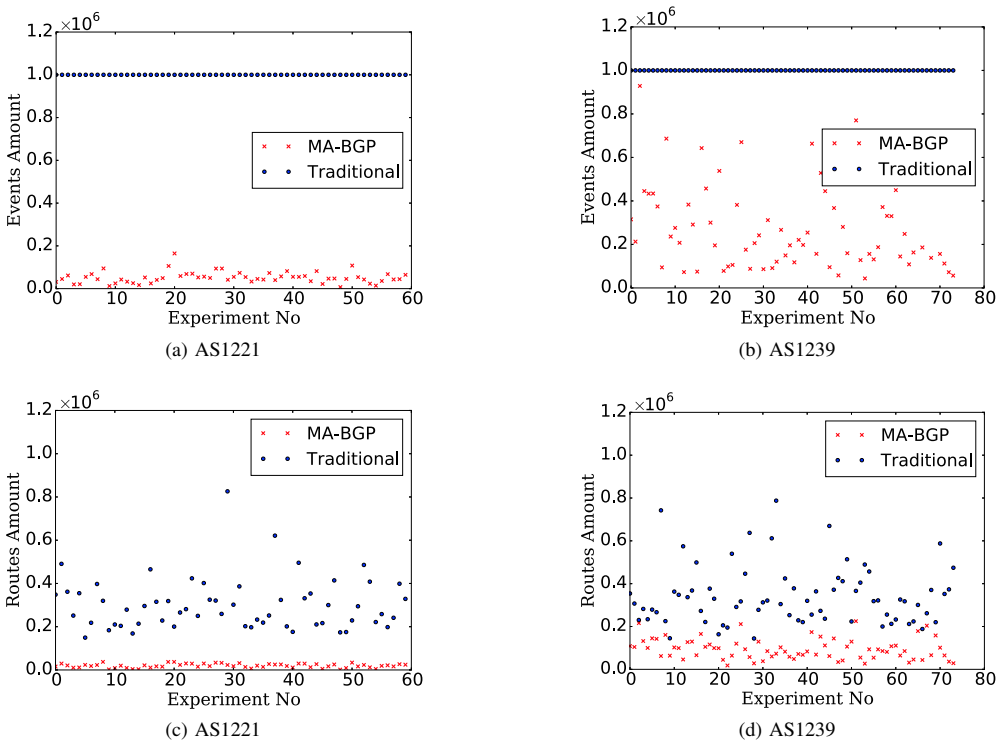
Fig. 7: Validity Evaluation. Single experiment on each AS.

- Large-scale realistic topologies.
  In this case, we download 6 realistic network configurations from Rocketfuel [16], including routers information and IGP topologies. However, iBGP topologies also need to be configured by our program, so there can be various experimental configurations with the same IGP topology but distinct iBGP topologies. The intention of tens of thousands of large-scale experiments with hundreds of routers is to prove that MA-BGP can also work in nearly realistic networks.

For each configuration, we simulate it in original C-BGP and MA-BGP, recording the total amount of events as well as routes separately. Events amount refers to the total number of BGP Open, Update, Notification and Keepalive packets. Routes amount means the total number of routes propagated in the network. According to the scale of the topology, a large enough corresponding upper bound of events amount is set by experience. The simulation will terminate when the number of events exceeds the upper bound, and then report an oscillation.

B. Evaluation

The evaluation can be divided into two aspects. On the one hand, we have to analyze the validity of MA-BGP, that is, check if all the topologies which oscillate in traditional C-BGP can converge to a stable routing state in MA-BGP. On the other hand, the overhead also needs to be estimated. To guarantee the stability, extra routes have to be saved in routers with oscillations and disseminated to the neighbors. We conduct tens of thousands of experiments.



(a) AS1221 (b) AS1239

(c) AS1221 (d) AS1239

Fig. 8: Validity Evaluation. 100 experiments are conducted for each AS and here are those results which oscillate in traditional BGP. (a) and (b) represent total events amount of AS1221 and AS1239 respectively while (c) and (d) are total routes amount.

Fig. 7 shows the results of single experiment on each of the 6 AS-es with certain iBGP topologies. The total events amount and routes amount are presented in (a) and (b) respectively. As shown in Fig. 7, topologies persistently oscillating (which cannot converge within the upper bound events in our experiments, such as AS1239) can converge to a stable state in MA-BGP, and stable topologies (which can converge after a finite events, such as AS1221) will still converge soon with a little overhead introduced. These results turn out that the validity of MA-BGP.

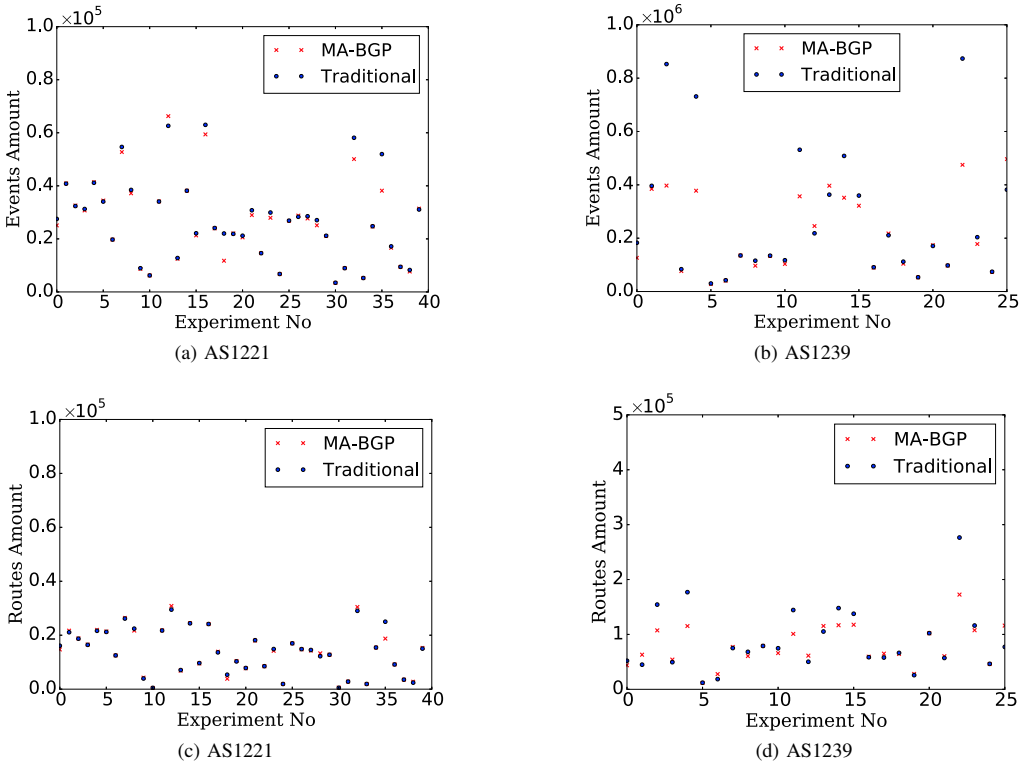
Fig. 8 depicts the comparison results of the oscillating-in-traditional-BGP topologies among 100 experiments on AS1221 and AS1239 respectively and Fig. 9 shows those converged-in-traditional-BGP ones. As shown in Fig. 8, those topologies oscillating in traditional BGP can converge to a stable routing state, and MA-BGP reduces the total events amount by 10% to more than 90% and reduces the total routes amount by 5% to more than 90%. It indicates that MA-BGP can eliminate iBGP routing oscillations effectively.

As Fig. 9 illustrates, those topologies converging in traditional BGP would not oscillate in MA-BGP as well, that is, no other routing problem is introduced. However, the sum of routing events and routes are not always the same. After elaborative observation, we come to the following conclusion. Transient oscillations are the reason why the topologies which converge in traditional BGP also change their routes dissemination process in MA-BGP. Timing coincidences, such as message delays or a particular order that routers send and receive messages, can lead to transient oscillations which disappear when the coincidences no longer exist.

A transient oscillation in a node behaves as limited times of changing it best route. This topology will eventually converge, though there would exist several unnecessary routing updates. If there is merely one iteration of procedure a and procedure b, one extra route would be disseminated as overhead, and the total events amount would stay the same with traditional BGP. If there are more than one iterations, one additional route would be added upon the first iteration and thus eliminate the following iterations. In this case, the total events and routes amount might both decrease compared to the traditional BGP. Nevertheless, there is no need to worry about new oscillations might be introduced due to the increasing visibility of paths for the nodes.

VII. Conclusion

Routing oscillation in BGP is a sticky challenge for decades. In this paper, we have proposed a new idea based on the Add-Paths mechanism named the Minimal Add-Paths BGP (MA-BGP) algorithm for solving iBGP stability problems with minimal transmission overhead introduced. First, we summarize a necessary and sufficient condition to detect inappropriate BGP updates which may cause routing oscillations. Furthermore, MA-BGP allows BGP routers to disseminate multiple paths in one update simultaneously. In this way, whenever an oscillation occurs, MA-BGP algorithm will be



(a) AS1221 (b) AS1239

(c) AS1221 (d) AS1239

Fig. 9: Communication Overheads Evaluation. 100 experiments are conducted for each AS and here are those topologies which converge in traditional BGP. (a) and (b) represent total events amount of AS1221 and AS1239 respectively while (c) and (d) are total route amount.

triggered and one extra path will be advertised by the router. Only single additional path needs to be advertised to break an oscillation for each BGP router, which could be classified to Add-p-Path ($p \leq 1$) Mode in Add-Paths Modes.

ACKNOWLEDGMENT

This work was supported by the National Natural Science Foundation of China under Grant 61572278 and 61502268, the National Key R&D Program of China under Grant 2016YFB0800102, and the National Basic Research Program of China (973 Program) under Grant 2012CB315803. We would also like to show our deep gratitude to Prof. Bruno Quoitin for his great efforts to timely answer our questions, and moreover, fix C-BGP to a new release (v2.3.2).

REFERENCES

[1] Y. Rekhter, T. Li, and S. Hares, "A Border Gateway Protocol 4 (BGP-4)," *RFC 4271*, 2006.

[2] V. Gill, D. McPherson, A. Retana, and D. Walton, "Border gateway protocol (bgp) persistent route oscillation condition," *RFC 3345*, 2002.

[3] T. G. Griffin and G. Wilfong, "On the correctness of iBGP configuration," *ACM SIGCOMM Computer Communication Review*, vol. 32, no. 4, pp. 17–29, 2002.

[4] A. Basu, C. H. L. Ong, A. Rasala, F. B. Shepherd, and G. Wilfong, "Route oscillations in I-BGP with route reflection," *ACM SIGCOMM Computer Communication Review*, vol. 32, no. 4, pp. 235–247, 2003.

[5] T. G. Griffin, F. B. Shepherd, and G. Wilfong, "The stable paths problem and interdomain routing," *IEEE/ACM Transactions on Networking (ToN)*, vol. 10, no. 2, pp. 232–243, 2002.

[6] T. G. Griffin and G. Wilfong, "Analysis of the med oscillation problem in bgp," in *Proceedings. IEEE International Conference on Network Protocols*. IEEE, 2002, pp. 90–99.

[7] A. Flavel and M. Roughan, "Stable and flexible iBGP," *ACM SIGCOMM Computer Communication Review*, vol. 39, no. 4, pp. 183–194, 2009.

[8] T. Bates, R. Chandra, and E. Chen, *BGP Route Reflection - An Alternative to Full Mesh iBGP*. RFC Editor, 2000.

[9] M. Vutukuru, P. Valiant, S. Kopparty, and H. Balakrishnan, "How to Construct a Correct and Scalable iBGP Configuration," in *Proceedings. IEEE INFOCOM*, 2006, pp. 1–12.

[10] M.-O. Buob, S. Uhlig, and M. Meulle, "Designing optimal ibgp route-reflection topologies," in *NETWORKING 2008 Ad Hoc and Sensor Networks, Wireless Networks, Next Generation Internet*. Springer, 2008, pp. 542–553.

[11] B. Quoitin and S. Uhlig, "Modeling the routing of an autonomous system with c-bgp," *IEEE network*, vol. 19, no. 6, pp. 12–19, 2005.

[12] S. Vissicchio, L. Cittadini, L. Vanbever, and O. Bonaventure, "iBGP deceptions: More sessions, fewer routes," in *Proceedings. IEEE INFOCOM*, 2012, pp. 2122–2130.

[13] D. Walton, A. Retana, E. Chen, and J. Scudder, "BGP Persistent Route Oscillation Solutions," *Internet-Draft*, 2014, draft-ietf-idr-route-oscillation-stop-02.txt, work in progress.

[14] R. Musunuri and J. A. Cobb, "A complete solution for ibgp stability," in *Proceedings. IEEE International Conference on Communications*, vol. 2. IEEE, 2004, pp. 1177–1181.

[15] V. Van, den Schrieck, P. Francois, and O. Bonaventure, "BGP Add-Paths: The Scaling/Performance Tradeoffs," *IEEE J. Sel. Areas Commun*, vol. 28, no. 8, pp. 1299–1307, 2010.

[16] N. Spring, R. Mahajan, and D. Wetherall, "Measuring ISP topologies with rocketfuel," *ACM SIGCOMM Computer Communication Review*, vol. 32, no. 4, pp. 133–145, 2002.