BGP WITH AN ADAPTIVE MINIMAL ROUTE ADVERTISEMENT INTERVAL

by

Nenad Lasković
B.Sc., University of Novi Sad, Novi Sad, Serbia and Montenegro, 1999

THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF

Master of Applied Science

In the School of Engineering Science

SIMON FRASER UNIVERSITY

Spring 2006

Approval

Name: **Nenad Lasković**

Degree: **Master of Applied Science**

Title of Thesis: **BGP with an Adaptive Minimal Route Advertisement Interval**

Examining Committee:

Chair: **Dr. Shahram Payandeh**
Professor, School of Engineering Science

_____

Dr. Ljiljana Trajković
Senior Supervisor
Professor, School of Engineering Science

_____

Dr. Stephen Hardy
Supervisor
Professor, School of Engineering Science

_____

Dr. Uwe Glässer
Examiner
Associate Professor, School of Computer Science

Date Defended/Approved: _____

Abstract

The duration of the Minimal Route Advertisement Interval (MRAI) and the implementation of MRAI timers have a significant influence on the convergence time of the Border Gateway Protocol (BGP). Previous studies have reported existence of optimal MRAI values that minimize the BGP convergence time for various network topologies and traffic loads. In this thesis, we propose the *adaptive MRAI* algorithm for adaptive adjustment of MRAI values. We also introduce *reusable MRAI timers* that limit the number of advertisements for each destination. The modified BGP is named *BGP with adaptive MRAI* (BGP-AM). BGP-AM performance is evaluated using the BGP processing delay based on reported measurements. ns-2 simulation results demonstrate that BGP-AM leads to a shorter convergence time and a number of update messages comparable to the current BGP. Furthermore, BGP-AM convergence time depends linearly on the BGP processing delay.

Dedication

Mami i Jasmini,
za svu vašu ljubav

To my mom and to Jasmina,
for all your love

Acknowledgements

I would like to thank my senior supervisor, Professor Ljiljana Trajković, for her guidance and support during my studies at the Communication Network Laboratory at Simon Fraser University.

I would also like to thank Dr. Stephen Hardy and Dr. Uwe Glässer for serving on my examining committee and Dr. Shahram Payandeh for chairing my thesis defense.

My special thanks go to Dr. Miroslav Despotović. His confidence in me and his tutoring have enabled me to overcome all difficulties during my research.

All my friends, here at SFU and all over the world have supported me during my studies at Simon Fraser University. They shared with me all my good and bad times. Unfortunately, the space does not allow me to mention them all. I will mention the ones with whom I have spent the most time in the last two years. Tony, thank you for your BGP implementation, without which I could not be able to finish my thesis. Nikola, thank you for friendship and all the laughs that we have had in the lab. Slobodan, thank you for bringing a big part of Novi Sad to Vancouver.

The most important person I left for the end. Jasmina, thank you for everything.

Table of Contents

List of Figures

List of Tables

Glossary

Adj-RIBs-In	set of RIBs for incoming routes from adjacent routers
Adj-RIBs-Out	set of RIBs for outgoing routes from adjacent routers
AS	Autonomous System
ATM	Asynchronous Transfer Mode
BGP	Border Gateway Protocol
BGP-4	Border Gateway Protocol version 4
BGP-AM	Border Gateway Protocol with adaptive MRAI
FDDI	Fiber-Distributed Data Interface
GF	Ghost Flushing
eBGP	External BGP

iBGP	Internal BGP
IDR	Inter-Domain Routing
IGP	Interior Gateway Protocol
IS-IS	Intermediate System to Intermediate System
ISP	Internet Service Provider
LAN	Local Area Network
Loc-RIB	RIB for locally used routes
MED	Multiple Exit Discriminator
MRAI	Minimum Route Advertisement Interval
NAP	Network Access Point
NLRI	Network Layer Reachability Information
NSP	Network Service Provider
OSPF	Open Shortest Path First
RCN	Root Cause Notification or Root Cause Node
RFC	Request for Comments
RFD	Route Flap Damping
RIB	Routing Information Base
RIP	Routing Information Protocol
RSP	Regional Service Provider

SSFNet	Scalable Simulation Framework Network models
SSF.OS.BGP4	SSFNet BGP

Chapter 1

INTRODUCTION

The Internet consists of numerous heterogeneous networks without a centralized control. These networks are clustered in groups called Autonomous Systems (AS), where each AS is controlled by a common administrative entity. Examples of ASs are Internet Service Providers and company/university campus networks. Communication between ASs requires a common protocol. Border Gateway Protocol (BGP) [32] is the de facto standard inter-domain routing protocol in today's Internet.

BGP suffers from long convergence time. The BGP convergence time is time elapsed from the moment when a change occurs in a network until all routers accordingly adjust their routing tables [11]. This updating of route information is called the BGP convergence process. During this process, routing tables may contain obsolete routing information, which may cause inaccessibility of ASs, packet loss, and additional overhead to routers [16], [17]. The goal of this thesis is to propose a minor modification

to BGP, which decreases its convergence time without changing the format of BGP messages or the BGP functionality.

To reduce the overall number of messages and the convergence time, BGP limits the rate of messages exchanged between routers. One of the rate limiting parameters is the Minimal Route Advertisement Interval (MRAI). MRAI limits the minimum time interval between two consecutive update messages sent for the same destination. The BGP convergence time is affected by the duration of MRAI and the implementation of MRAI timers. The default MRAI value (30 s) is used in the majority of today's routers [15]. This value is not optimal for every network topology and using smaller values may lead to a significant decrease of the BGP convergence time [11]. It has also been reported that an optimal MRAI value depends on the network topology and traffic load [11]. Adaptive MRAI timers were proposed as one solution for finding an optimal MRAI value [17]. However, the implementation details have not been presented. Therefore, our motivation was to find a solution that would replace one global MRAI value and improve BGP convergence time.

In this thesis, we propose an *adaptive MRAI* algorithm for adjusting MRAI values for every destination in each BGP router. The current implementation of MRAI timers (*per-peer MRAI timers*) prolongs the BGP convergence time because they impose delay on all route advertisements regardless of their destinations. Hence, we propose using *reusable MRAI timers* that independently limit advertisements of distinct destinations, while retaining the efficiency of per-peer MRAI timers. The proposed BGP modification, named *BGP with adaptive MRAI* (BGP-AM), employs the adaptive MRAI algorithm and reusable MRAI timers.

An accurate estimation of the BGP processing delay (the delay due to processing of BGP messages in routers) is important for analysis and simulation of the BGP dynamic behavior. A widely used approach for calculating the BGP processing delay (named here the *uniform* BGP processing delay) assumes that the average time needed for processing BGP messages depends linearly on the number of received messages [11]. However, recent measurements [7] indicate that using the uniform delay leads to unrealistically high BGP processing delay estimates. We used the *empirical* BGP processing delay based on measurements [1], [7].

We have implemented BGP-AM and both the empirical and uniform delays in the ns-2 simulator [24]. Simulation results indicate that BGP-AM leads to a shorter convergence time with a comparable number of update messages as in the current BGP [14].

The remaining of the thesis is organized as follows. Chapters 2 and 3 describe BGP and its dynamic behavior. The previous work on improving BGP convergence time is presented in Chapter 4. In Chapter 5, we describe BGP-AM and the empirical BGP processing delay. The implementation of BGP-AM and simulation results for various network topologies are presented in Chapter 6. We conclude with Chapter 7.

Chapter 2

BORDER GATEWAY PROTOCOL (BGP)

Routing protocols in the Internet are divided in two groups, *intra-domain* and *inter-domain* protocols. Intra-domain or Interior Gateway Protocols (IGP) are used for routing within ASs. Examples of IGPs are Routing Information Protocol (RIP), Open Shortest Path First (OSPF), and Intermediate System-to-Intermediate System (IS-IS). The administrator of an AS may choose any IGP protocol, because these protocols do not affect the systems outside an AS. To the contrary, one common inter-domain protocol is needed for communication between ASs. The standard inter-domain routing protocol in today's Internet is Border Gateway Protocol (BGP), defined in the RFC 1771 [32]. The relationship between intra-domain and inter-domain routing protocols is shown in Figure 1. BGP routers are called *BGP speakers*. Two neighboring BGP speakers that exchange routing information are called *peers*.

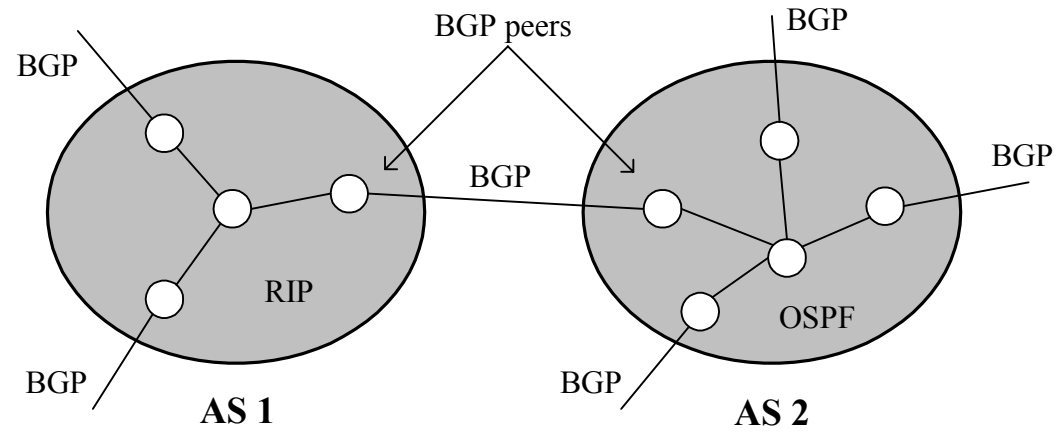

Figure 1. An example of inter-domain and intra-domain routing protocols.

ASs usually have more than one BGP speaker because one BGP speaker is not sufficient to accomplish all the inter-domain routing tasks of one ASs. BGP speakers within one AS also communicate using BGP, which is then referred to as Interior BGP (iBGP). When BGP is used between BGP speakers that belong to different ASs, it is called External BGP (eBGP). To ensure unambiguous communication, each AS is identified with a unique 16-bit number, called AS number, assigned in a similar manner as IP addresses.

2.1 AS Hierarchy

The topology of ASs is loosely hierarchical and divided into three levels, from Tier I (the highest level) to Tier III (the lowest level) as shown in Figure 2 [4]. Tier I ASs are Network Service Providers (NSP) that create the backbone of the Internet, Tier II ASs are Regional Service Providers (RSP), and Tier III ASs are Internet Service Providers (ISP) that provide Internet access to individual subscribers. NSPs are interconnected

through Network Access Points (NAP), which are typically Local Area Networks (LAN), such as Ethernet, Asynchronous Transfer Mode (ATM), or Fiber-Distributed Data Interface (FDDI).

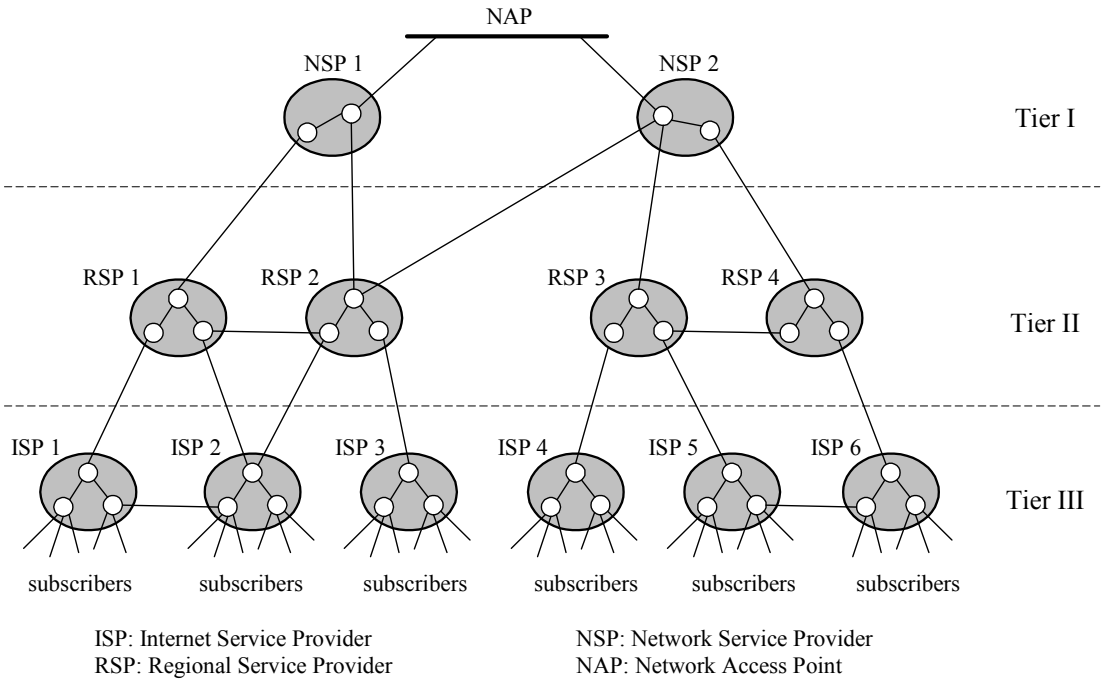

Figure 2. An example of three layer AS hierarchy of the Internet and relationships between ASs.

The relationship between two ASs on different levels is based on their commercial agreements, where one AS (customer) pays for services of another AS (service provider). ISPs of Tier III are customers of Tier II regional providers, whereas regional service providers of Tier II are customers of Tier I network service providers. Examples of a customer-provider agreement are the relationships between ISP 1 and RSP 1 and between RSP 2 and NSP 2 (Figure 2). These types of relationships usually have asymmetric traffic loads. The traffic from the service provider to the customer requires higher data rate than the traffic in the opposite direction.

The second type of relationship between ASs is a peering agreement between ASs that belong to the same Tier level. In this case, the traffic in both directions is balanced. As both ASs provide and use services simultaneously, neither of them pays. Examples of peering agreements are the relationships between ISP 1 and ISP 2 and between RSP 3 and RSP 4 (Figure 2).

2.2 Exchange of Routing Information in BGP

BGP speakers perform two tasks. The first task is forwarding of packets between end systems on the Internet. Each end systems on the Internet is defined by a unique 32 bit IP address. All end systems that belong to one network have identical first *n* bits of their IP addresses. These bits uniquely define a single network. They are called network address or IP address prefix. To forward a packet, a router needs to know only the network address of the destination end system. Once the packet reaches the designated network, the router of that local network forwards it to the particular end system. Therefore, for a router the destination of a packet is a network, rather than a single end system. For the same reason, routers in their routing tables store only network addresses, called destinations. One destination in a BGP routing table is represented by a pair consisting of an IP address prefix and the length of the prefix, as shown in Figure 3.

The second task of BGP speakers is maintaining information regarding routes (paths) from a particular speaker toward destinations in the Internet. The path contains a list of AS numbers, which describes all ASs which a packet has to traverse along the route to the destination. A BGP speaker may store multiple paths to each destination. Those paths are stored in a BGP routing table or RIB (Routing Information Base). The list of AS numbers in a path conveys more information than the distance used in

traditional distance-vector protocols. It is used to prevent creation of routing information loops. That is also the reason why BGP is often called *path-vector* protocol, to distinguish it from distance-vector routing protocols. BGP uses the length of a path (the number of ASs in the path), as a distance metric. If there are several routes to one destination, the shortest route is called the best route and it is used for forwarding packets. All other alternative routes may be considered as back-up routes in the case that the best route becomes unfeasible.

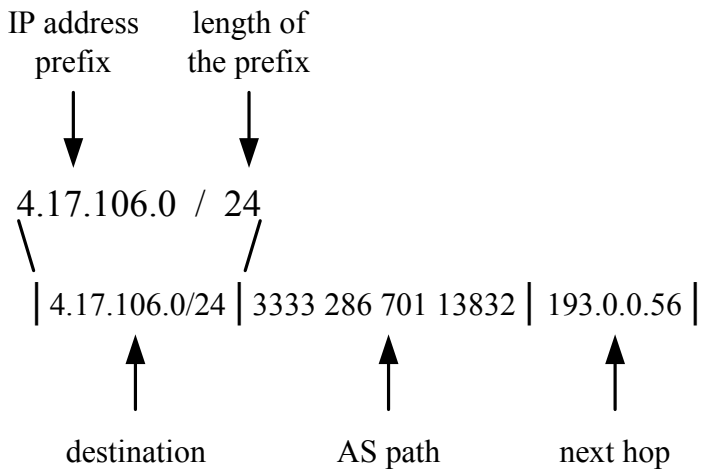

Figure 3. One simplified record in BGP routing table. The AS path and the next hop are given for each destination.

Due to commercial reasons, ASs do not advertise all their routing information to the entire Internet. To control exchange of the routing information, BGP implements routing policies [32] that override the distance metric. A previous study [36] has shown that the routing policies may be selected in such a manner to results in persistent route oscillations. Those conflicts in routing policies are not encountered in practice.

BGP exchanges routing information only between peers, using four types of messages: *open*, *update*, *notification*, and *keep-alive*. Update messages are used to

exchange routing information, while the remaining three types are used to handle connections between BGP peers. Once two BGP speakers have established a connection, they exchange their best routes to all destinations. After the initial exchange of routing information, BGP peers perform incremental updates only when changes in the Internet topology cause a replacement of one of their best routes. BGP speakers convey information regarding new best routes using two types of BGP update messages. The first type is a *route advertisement* that contains announcement of a new best path to a destination. A route advertisement may introduce a path to a new destination or a new best route to the previously advertised destination. The second type of a BGP update message is a *route withdrawal* (or *explicit withdrawal*) that declares that the previously advertised best route to a destination became unavailable. A BGP speaker that sends a withdrawal does not have any path to the related destination.

A route advertisement announcing a new best path to a previously advertised destination is called an *implicit withdrawal*. This type of an advertisement is considered to be similar to a withdrawal because it may cause replacement of the best route in another BGP speaker.

2.3 BGP Update Message Format

BGP messages are conveyed in TCP segments using TCP port 179 [32]. The maximum segment size is 4069 octets (bytes). All types of BGP messages have an identical, fixed-size header that contains: a *marker* for detecting a loss of signalization, the *length* of message in octets, and the *type code* of the message. The type code of update messages is 2.

Update messages carry routing information that BGP speakers use to determine paths to various ASs. One update message may contain a single feasible route to a group of destinations and a list of unfeasible routes. The format of an update messages is shown in Table 1.

Unfeasible Routes Length (2 octets)
Withdrawn Routes (variable)
Total Path Attribute Length (2 octets)
Path Attribute (variable)
Network Layer Reachability Information – NLRI (variable)

Table 1. BGP update message format.

The *Unfeasible routes length* field specifies the overall length of the *withdrawn routes* field. Zero length indicates that no routes are being withdrawn. The withdrawn routes field contains a list of destinations that have become unreachable. Implicitly, by listing unreachable destinations, a BGP speaker announces that it does not have any path to them.

The *Total Path Attribute Length* field specifies the overall length of the *path attribute* field. A zero length indicates that no routes are being advertised. Path attributes describe details of the advertised path, such as: the origin of the path (IGP, EGP, or incomplete), the list of ASs (AS path), the next hop, and local preferences of the path. NLRI is a list of destinations reachable by using the advertised path.

A simplified example of a BGP update message is shown in Table 2. The first two rows indicate the IP address and the AS number of the sender and receiver of the message, respectively. A BGP speaker may withdraw several previously advertised routes by listing unreachable destinations in the withdrawn routes field. BGP messages

may advertise a single route specified by the AS path. However, that route may lead to more than one destination listed in the NLRI field.

From:	208.51.113.254 (AS 3549)
To:	198.32.162.102 (AS 6447)
Withdrawn routes (destinations):	213.193.32.0/24 213.193.48.0/24
AS path:	3549 701 2712
Advertised destinations (NLRI):	134.132.250.0/24 134.133.46.0/24 134.138.181.0/24

Table 2. Simplified BGP update message containing a list of withdrawn destinations, an AS path, and a list of advertised destinations associated with the AS path.

Chapter 3

DYNAMIC BEHAVIOR OF BGP

Due to persistent changes of the Internet topology, BGP speakers need to exchange a large number of update messages. Based on these messages, each BGP speaker adds new and deletes unfeasible routes to destinations. Therefore, BGP is characterized by continuous transformations of routing tables. In this thesis, we are especially interested in this dynamic aspect of BGP.

Handling update messages may be divided into two steps: the *Decision Process* and the *Update-Sent Process* [32]. When a BGP speaker receives an update message, the first step is the Decisions Process. In this step, a BGP speaker has to check whether that message affects any of the best routes. The duration of the Decision Process is called BGP processing delay. If the received update message causes the routing table to change, the BGP speaker passes to the Update-Sent Process and sends updates to its peers. In the case that there are multiple paths to a destination, the BGP speaker may need to perform

several iterations of exchanging messages with its peers, until it finds the best route, i.e., until it converges. This process of updating routing tables after a network change is called the BGP convergence process. The end of the BGP convergence process for a single destination is defined as the moment when all BGP speakers in a network converge and when stop to generate update messages regarding the destination. However, particular BGP speakers may converge before the BGP convergence process has been completed for the entire network. For example, BGP speakers closer to the origin of the network change may learn the best route earlier than other BGP speakers.

The duration of the BGP convergence process is called BGP convergence time. For a single destination, it may be defined as the time elapsed from the instant when the first update message containing a change of the destination reachability is sent until all update messages that are a consequence of the original update are received [11].

3.1 MRAI Timers

A BGP speaker may learn about a change of destination reachability from multiple peers. Due to the variation of propagation times, peers may have different best routes to a destination. A BGP speaker selects the best route from all received routes. A BGP speaker may also receive a number of suboptimal routes before receiving the best route. A previous research [11] shows that if a BGP speaker responds to received updates instantaneously by sending updates to its peers, the number of update messages and BGP convergence time would increase. A BGP speaker cannot wait indefinitely to receive the best route. Hence, it has to minimize the number of update messages and react in a timely manner to changes in the Internet topology. A solution, proposed in RFC 1771 [32], is rate limiting: it limits the frequency of route advertisements by imposing a minimal

interval of time that should pass between two consecutive advertisements of the same destination sent from a BGP speaker to one of its peers. This interval is called the Minimal Route Advertisement Interval (MRAI) or the MRAI round. In the case of multiple paths to a destination, several MRAI rounds may be needed until the best route is found and convergence achieved. The rate limiting is applied only to advertisements between neighboring ASs and it does not affect route advertisements within an AS. Furthermore, withdrawal rate limiting (WRL) is not applied because it leads to an increase of BGP convergence time [11], [31]. WRL has not been endorsed by RFC 1771 [32] and it is not used in the majority of routers [15].

RFC 1771 [32] specifies the duration of an MRAI round to be 30 s, which is controlled by using MRAI timers. However, manufacturers may use different values for the duration of MRAI round. For example, Juniper's default configuration sets MRAI to 0 s [7], [20]. To avoid synchronization and possible peaks in the update messages distribution, RFC 1771 [32] proposes using values of MRAI multiply by a uniform jitter in the range 0.75 – 1. Nevertheless, the majority of BGP speakers in the Internet do not implement this MRAI modification [15], [31].

The independent rate limiting of various destinations may be achieved by using per-destination MRAI timers, where one per-destination MRAI timer is associated with one destination in a routing table. The routing table in a core Internet router contains over 100,000 destinations [31] and the implementation of such a large number of timers is not feasible. Hence, RFC 1771 [32] proposes implementing per-peer, rather than per-destination, MRAI timers: one per-peer MRAI timer is associated with one peer. The timer is set when a route advertisement is sent to the corresponding peer, regardless of its

destination. An advantage of per-peer timers is that their number is equal to the maximum number (several hundred) of peers of one BGP speaker. A disadvantage is that they affect not only the route advertisements that were sent in the last MRAI round, but also all advertisements of new destinations that will be sent to the peer.

An illustration of per-destination and per-peer MRAI timers for a simple network with four ASs is shown in Figure 4. AS3 receives advertisements of destinations 1 and 2 within a short time interval (for example 1 second). MRAI timers are not set. The first update is sent from AS1 to AS3 with advertisement 1. AS3 receives advertisement 1 at time t_{R1} and, after processing it, sends it to AS4 at time $t_{S1,}$ as shown in Figure 4(a). AS3 receives advertisement 2 from AS2 at $t_{R2} = t_{S1}+1$ second. The sending time t_{S2} of advertisement 2 from AS3 depends on the implementation of MRAI timers. If per-destination MRAI timers are used, AS3 sends advertisement 2 to AS4 at time t_{S2}, immediately after processing it, as shown in Figure 4(b). (Sending advertisements of different destinations is independent.) However, if per-peer MRAI timers are used, advertisement 1 sets the per-peer MRAI timer associated with peer AS4 and, thus, sending all subsequent advertisements to AS4 will be delayed. Hence, AS3 has to postpone sending advertisement 2 until the end of the MRAI round, as shown in Figure 4(c). As a result, advertisement 2 to AS4 is delayed by 29 s.

When there is an extensive exchange of advertisements between BGP speakers, a per-peer MRAI timer is never idle, because a new advertisement would cause the timer to start as soon as it has expired. As a result, a per-peer MRAI timer may be viewed as a continuous timer with a 30 s cycle. Hence, the average delay of one route advertisement per BGP speaker is half of one MRAI round, as observed in measurements of Internet

traffic [17]. The BGP processing delay does not affect the average delay, because is considered negligible compared to the duration of an MRAI round.

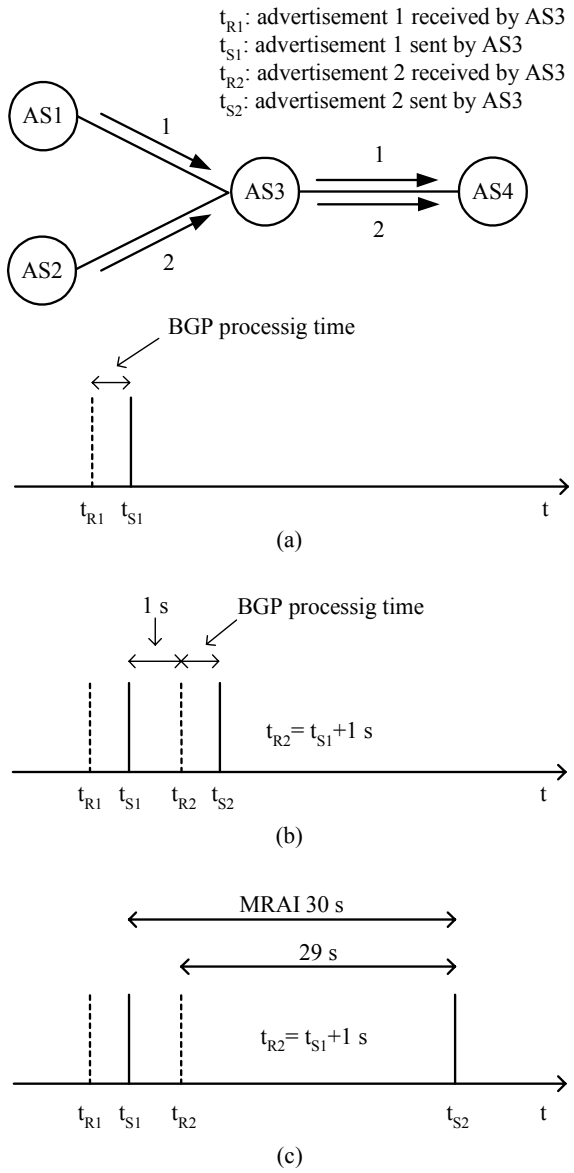

Figure 4. Propagation of BGP update messages. AS3 receives update from AS1 and sends a new update to AS4 (a). After 1 second AS3 receives an update from AS2. When per-destination MRAI timers are used AS3 can send update to AS4 immediately (b), whereas in the case of per-peer MRAI timers AS3 has to wait until the end of MRAI period (c).

A path in a network is defined in order to estimate time needed for a route advertisement to traverse a route between BGP speakers. A simple path *p* is a sequence of

k nodes (u_1, …, u_k), such that each node is included in the path only once (a path cannot have loops) [17]. In the case of per-peer MRAI timers, time $t(p)$ required for a route advertisement to traverse path p is:

$$t(p) \approx |p| \times \frac{MRAI}{2}, \tag{1}$$

where $|p|$ is the number of hops of path p. Note that we assume that all BGP speakers have identical MRAI, i.e., that it is not modified by a uniform jitter.

The maximal time that a route advertisement may be delayed in one BGP speaker is equal to MRAI and, hence, the upper bound for $t(p)$ is:

$$t(p) \leqslant |p| \times MRAI . \tag{2}$$

3.2 BGP Convergence Time

BGP convergence time captures the ability of BGP to adjust to changes in the Internet topology. Commonly used analysis [11], [31] and measurement [16]–[18] scenarios consist of two phases: *up* (advertisement) and *down* (withdrawal) phase. In the up phase, a new destination is introduced to a network. The destination is directly connected to a single BGP speaker called the *origin*. The convergence time T_{up} is the time between the instant when the first update message is sent from the origin until all BGP speakers find the shortest path to the destination. At the end of the up phase, the origin sends a withdrawal of the destination. This marks the beginning of the down phase. The convergence time T_{down} is the time needed for all BGP speakers to reach the new steady-state when they have no path to the destination (the origin is the only BGP speaker connected to the destination). Although more complicated scenarios may occur in the

deployed networks than the scenario with two distinct up and down phases, it represents two characteristic cases of the BGP convergence process.

In the *up* phase, the best route to the new destination for each BGP speaker is the shortest path to the origin. Hence, the convergence time of a BGP speaker depends on the minimal distance (measured by the number of hops) to the origin. Therefore, convergence time T_{up} depends on the BGP speaker farthest from the origin. The farthest BGP speaker is the BGP speaker with the longest shortest path to the destination. When per-peer MRAI timers are used, the convergence time T_{up} is estimated as (1):

$$T_{up} \approx \left| p_{short} \right| \times \frac{MRAI}{2}, \tag{3}$$

where $|p_{short}|$ is the length of the shortest path from the destination to the farthest BGP speaker [17]. Based on (2), the upper bound of the convergence time T_{up} is:

$$T_{up} \leqslant \left| p_{short} \right| \times MRAI \ . \tag{4}$$

The convergence time T_{up} depends on the network diameter and the average delay of an update message in a BGP speaker. The network diameter is a constant and cannot be reduced without altering the network topology. However, the average delay of updates in BGP speakers is affected by the implementation of MRAI timers and the duration of the MRAI round.

At the beginning of the down phase, the origin sends withdrawals of the destination to all its peers. The peers propagate this information further through the network. After BGP speakers learn that the best route has become unfeasible, they try to replace it with the next best alternative route. If there is more than one alternative route,

BGP speakers may need to explore all possible paths to the destination. For each path tried as the best route, a BGP speaker needs to send a route advertisement. After the first advertisement is sent, MRAI timers are set, delaying sending other advertisements. Hence, BGP speakers need several MRAI rounds to converge.

The duration of the convergence time T_{down} depends on the network topology and in particular, on the number of different paths to the destination. T_{down} also depends on the order in which BGP speakers explore paths. The worst case scenario is when BGP speakers need to explore the entire set P of all paths from the origin to all other BGP speakers. Therefore, the upper limit of the convergence time T_{down} is a function of the longest path in P [17]:

$$T_{down} \leqslant \max_{p \in P} t(p). \tag{5}$$

The convergence time T_{down} is also a function of the implementation of MRAI timers.

3.3 Uniform BGP Processing Delay

An important parameter in analysis of the BGP dynamic behavior is the delay imposed on an update message in a BGP speaker (BGP processing delay). It includes the queuing time of a message and the time needed for BGP to process the received message.

The uniform BGP processing delay [11], [31] is widely used in studies of BGP convergence time [3], [26], [28] – [30]. It has been implemented in SSFNET [34]. It assumes that a BGP speaker processes each update message independently. (Updates are processed one by one: while one update is processed, all others have to be queued.) Therefore, processing of one message is affected by the processing of other previously queued messages, as shown in Figure 5. The processing delay of each message is

estimated using a uniformly distributed random variable from the interval [p_{min}, p_{max}]. Commonly used values are p_{min} = 0.01 s and p_{max} = 1 s [11]. The average processing delay $t_{BGPprocess}(p)$ for a group of N queued updates is:

$$t_{BGPprocess}(p) = N \times \frac{[p_{\max} - p_{\min}]}{2}. \quad (6)$$

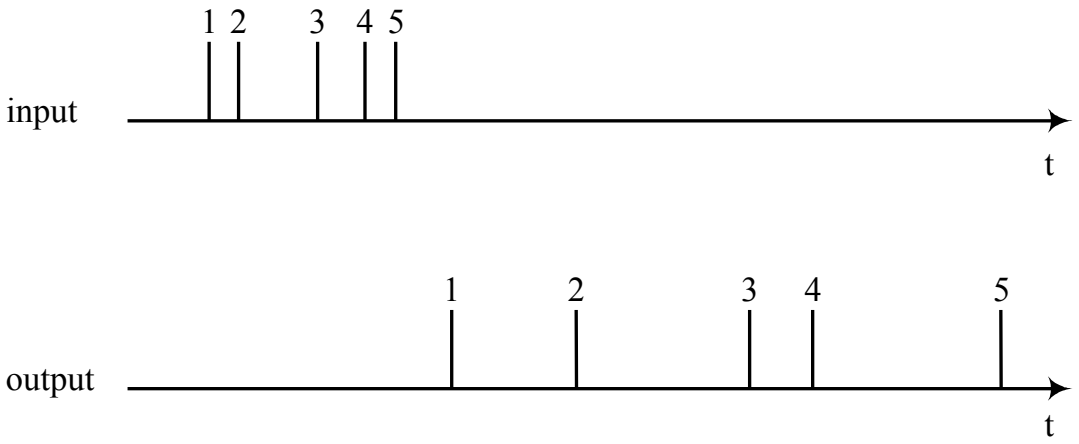

Figure 5. The uniform BGP processing delay. The delay of a message depends linearly on the number of previously received messages.

It follows from (6) that the average processing delay depends linearly on the number of update messages. Measurements have shown that this number in the core Internet routers may exceed 100 messages per second [31]. This suggests that the uniform BGP processing delay with an average processing delay of 0.5 s may not be appropriate for cases with extensive exchange of update messages. Even for a moderate number of messages (~20), the uniform BGP processing delay leads to unrealistically high values (~10 s) for the average BGP processing delay.

Recent measurements performed on Cisco routers [7] have indicated that the average BGP processing delay is much shorter than predicted by the uniform BGP

processing delay. The measurements revealed that the BGP speakers process groups of update messages in constant 200 ms processing cycles, as shown in Figure 6. When a BGP speaker operates below its maximum CPU utilization, it can process most messages that it receives at the end of a 200 ms cycle. The average BGP processing delay is between 101 ms and 110 ms. More than 95% of messages are processed within 210 ms. Nevertheless, in the case of high traffic loads, a BGP speaker cannot process all received messages in one 200 ms cycle and the maximum BGP processing delay may last several seconds.

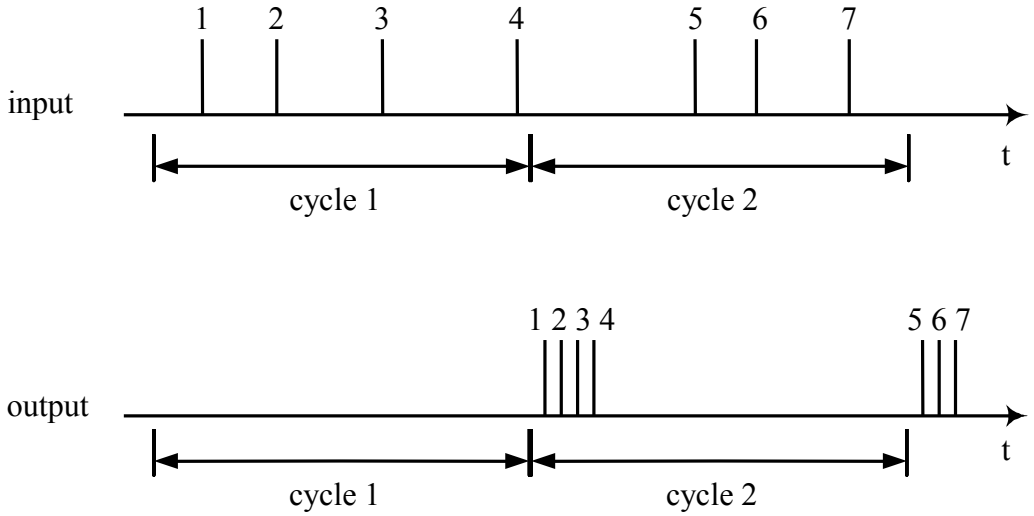

Figure 6. The empirical BGP processing delay. All messages received during a cycle are processed and sent at the end of the cycle.

Measurements of router CPU utilization [1] have revealed that core Internet routers operate under 50% of their capacity over 99% of time. The processing of BGP messages usually has a higher scheduling priority than any other process in a BGP speaker [7]. Hence, most of the time the routers process all the update messages within the 200 ms processing cycle. Although the implementation of BGP in routers of other

manufacturers may be different from the one reported in [7] for Cisco routers, we may conclude that the uniform BGP processing delay overestimates the delay experienced in deployed routers.

Chapter 4

PREVIOUS RELATED WORK

The growth of the Internet in the 1990s increased the interest for analysis of BGP and its dynamic behavior under these more demanding conditions. Both measurements [8], [16]–[20], [38] and theoretical analysis [6], [12]–[14], [33], [36] revealed that BGP had several issues regarding the convergence process. These issues can be divided in two groups. The first group consists of short-term and long-term instabilities, which may prolong BGP convergence time. The second group of issues may occur due to policy disputes, which may cause BGP speakers to oscillate between two or more possible routes to a destination. There is no a single solution for all these issues and previous studies usually concentrated on only one of them.

4.1 Short-term BGP Instabilities

The short-term BGP instabilities are caused by changes in network reachability, such as emergence of a new network or a change in an AS's routing policies. They may

delay the convergence by 3 to 15 minutes [16], causing routing loops [29] and loss of connectivity for large portions of the Internet. In this thesis, we are interested only in short-term BGP instabilities.

BGP convergence time in the case of short-term instabilities depends mainly on the implementation of MRAI timers and the ability of BGP to eliminate invalid (obsolete) routes [3], [28], [30]. Griffin and Premore [11] discovered that for each network topology, there were two MRAI values that minimized BGP convergence time and the number of exchanged update messages. These two MRAI values are similar for one network topology, whereas they vary for different topologies. For all simulated topologies [11], the optimal values of MRAI are less than 10 s, which is shorter than the default value for MRAI of 30 s specified by RFC 1771 [32]. The difference between the optimal and the default values of MRAI results in significantly longer than optimal convergence time. Furthermore, using MRAI values shorter than the optimal also leads to a manyfold increase of BGP convergence time and the number of update messages.

The studies [3], [26], [28], [30] following Griffin and Premore's research agreed that the duration of the MRAI round had a dominant influence on BGP time. They also concluded that one global MRAI value could not be determined for the entire Internet. The first reason is that various parts of the Internet require different optimal MRAI values. The second reason is that optimal MRAI values depend on the BGP processing delay that varies over time. Therefore, instead of changing the implementation of MRAI timers, researchers considered alternate approaches to decrease BGP convergence time.

During the convergence process, BGP explores all possible routes from a BGP speaker to a destination. BGP convergence time would be longer than the optimal if BGP

had to explore invalid or obsolete routes. Invalid routes are defined as routes that do not reach the destination. They are a result of transient information in BGP routing tables. A BGP speaker has an invalid route to a destination from the instant when the destination reachability changes until it learns the new best route. The existence of invalid routes has a particularly significant influence on BGP convergence time in the *down* phase. The withdrawal of the only route from the origin to the destination makes all other routes to that destination invalid.

The following three studies [3], [28], [30] propose solutions for eliminating the propagation of invalid routes in the Internet. The improvement of BGP convergence time for each of these three solutions depends on the network connectivity. Furthermore, these solutions increase the memory requirements and computational overhead of BGP speakers.

4.1.1 BGP with Consistency Assertions

Pei et al., [30] proposed BGP *with consistency assertions*, an enhancement that allowed BGP speakers to identify invalid routes. To detect invalid routes, the proposed enhancement compares advertisements from different neighbors. If a BGP speaker receives advertisements with different AS paths between two BGP speakers, it concludes that one or both advertisements are invalid. For example, two advertisements regarding a single destination in AS5 may be sent from two ASs. AS1 may send path (AS1, AS2, AS5) and AS2 may send path (AS2, AS3, AS4, AS5). These two ASs advertise different paths between AS2 and AS5. BGP *with consistency assertions* chooses the advertisement from the AS2 because it is more probable that AS2 has the accurate route from AS2 to AS5 than any other BGP speaker. On the contrary, based on BGP specified in RFC 1771

[32] the shortest path should be chosen, even it comprises the invalid route advertised by AS1. The reason that AS1 has the invalid route is that it uses the route learned from AS2 before the topology change.

The first drawback of the proposed solution is that it depends on the network topology and that for certain topologies it cannot detect invalid routes [28]. The second drawback is that it introduces computational overhead for comparing received advertisements from different peers. That overhead may be significant for networks with a large number of connections. The third drawback is that it requires sending extra information in BGP update messages [3].

4.1.2 BGP with the Ghost Flushing (BGP-GF)

Bremler-Barr et al., [3] proposed an algorithm that enables BGP speakers to inform their peers when routes become invalid. They named the advertisements containing invalid routes *ghost information* and the proposed algorithm the *ghost flushing* (GF). When a BGP speaker changes the best routes to a destination, it has to send advertisements regarding the change to its peers. However, sending the advertisements may be delayed due to MRAI timers. If the advertisements are delayed, the peers will have invalid routes that may be propagated further into the network. To avoid the delay, GF uses withdrawals to immediately send information that a route has become invalid. These withdrawals are called *flush* withdrawals and they have the same format as the withdrawals defined in RFC 1771 [32]. While the standard withdrawals are used to announce that a BGP does not have a route to a destination, flush withdrawals are used to eliminate an invalid route before the BGP speaker advertises the new best route. When MRAI timers expire, the BGP speaker will send the new best route and the peers will set

their routing tables accordingly. In other words, the ghost flushing creates a flood of withdrawal messages, which deletes ("flushes") all invalid routes ("ghosts") in the network and prepares the network for the new valid routes.

BGP with the ghost flushing (BGP-GF) has several drawbacks [30]. The first drawback is that BGP-GF may not eliminate all invalid routes, because of different propagation delays. The second drawback is that BGP-GF may increase the overall number of update messages, which may lead to suppression of a route due to route flap damping mechanisms [21], [38], described in Section 4.2. The third drawback is that BGP-GF requires additional memory for storing the last update message sent to each peer.

4.1.3 BGP with Root Cause Notification (BGP-RCN)

Pei at al., [28] proposed a new version of BGP, named BGP *with Root Cause Notification* (BGP-RCN). The main modification employed in BGP-RCN is that update messages convey additional information that specifies the cause of the route change. The additional information contains the address of the node that first detected the change, called the root cause node (RCN), and a sequence number. The RCN increments the sequence number every time it sends an update message. The RCN address and the sequence number are not modified by other BGP speakers and they are used to specify the origin of the route change and the time when the change occurred. Based on of this additional information, BGP speakers are able to detect and eliminate previously sent routes from the RCN. For example, in the down phase, all BGP speakers learn immediately from the first advertisement that a destination is no longer reachable, because the RCN had the only route to the destination. As a result, invalid routes are

efficiently eliminated from the network and BGP convergence time is considerably reduced than in the case of the current BGP [32].

The main drawback of BGP-RCN is that it changes the BGP update packet format. The second drawback is that for each route in the routing table, BGP-RCN stores the address of the RCN and the sequence number. The third drawback is that the performance of BGP-RCN depends on the network topology, particularly on BGP speakers' connectivity. The improvement of BGP-RCN over the current BGP is more significant when a topology change occurs near a BGP speaker with a smaller number of peers (Tier III) than near a BGP speaker that is a part of the Internet backbone (Tier I).

4.2 Long-term BGP Instabilities

Long-term BGP instabilities ("route flaps") are caused by persistent oscillations of unstable routes, generated by physical problems on data links or BGP speakers' configuration errors. BGP implements route flap damping (RFD) [37], a mechanism to suppress propagation of misbehaving routes. RFD specifies parameters that determine routes that should be suppressed and when the suppressed routes may be advertised again. The maximum interval that one route can be suppressed in one BGP speaker is one hour. Recent results by Mao et al., [21] showed that the original RFD mechanism may suppress valid routes and, hence, may deteriorate BGP performance. For example, a false detection of route flaps may occur in the down phase in networks with high connectivity. During route exploration, BGP generates a large number of messages, which may result in suppression of the route. Hence, Mao et al., [21] proposed the selective RFD that stored information regarding previously received routes and employed it to distinguish real route flaps from valid route exploration.

4.3 Policy Disputes in BGP

Routing policies enable BGP speakers to choose routes based on their metrics and also on the commercial relationships with their peers. Policies override distance-based metrics and each AS defines polices independently of other AS. The policy dispute is a situation in which policies in distinct ASs are in conflict, causing BGP to diverge [36].

The first proposed solution is presented as an abstract model of BGP, a Safe Path Vector Protocol (SPVP). Its implementation would guarantee BGP convergence [12]–[14], [27]. The SPVP analyzes policies between distinct ASs and does not permit policies that would lead to BGP divergence.

The second solution was proposed by Gao and Rexford [10]. They proposed a set of rules, which governed implementing policies in BGP speakers and guaranteed convergence of BGP. The hierarchical structure of the Internet AS topology and the commercial relationships between ASs is used to determine policies that would not cause in policy disputes.

Chapter 5

BGP WITH ADAPTIVE MRAI

The goal of this thesis is to investigate the effect of the advertisement rate limiting on BGP convergence time. Our results confirm previous findings [11] that an optimal MRAI value exists for each network topology. When is used, it minimizes BGP convergence time. Furthermore, we propose an algorithm for adaptive adjustment of MRAI value during the BGP convergence process. We also present a new approach for estimating the BGP processing delay.

5.1 Empirical BGP Processing Delay

We propose to use an empirical value for the BGP processing delay. It is based on the reported measurements of CPU utilization of BGP routers [1] and the average BGP processing delay [7].

We assume that BGP speakers operate under a usual traffic load and that they process update messages within 200 ms cycles. We also assume that a BGP speaker completes processing all received updates at the end of the 200 ms processing cycle. As a result, the processing delay is independent of the number of updates and the maximum processing delay of one update message is 200 ms. However, BGP speakers under heavy traffic load cannot process all received updates in one 200 ms processing cycle, which leads to longer processing delays [7]. This behavior of BGP speakers may be modeled by using longer processing cycles, which would result in longer processing delays.

To illustrate differences between the empirical and uniform BGP processing delays, we consider a simple case when MRAI timers are synchronized, as shown in Figure 7. In this example are used per-destination MRAI timers, BGP convergence process is in the down phase. We assume that one BGP speaker receives at most 20 messages in one MRAI round regarding a single destination. Shown are times when a BGP speaker receives (dashed lines) and sends (solid lines) update messages containing a single destination. The instant when a speaker sends update messages marks the beginning of an MRAI round. Each MRAI round consists of an *active* and an *idle* period.

The active period is the segment of the MRAI round when a BGP speaker processes the received update messages. It lasts from the beginning of an MRAI round until the last message in the round has been received. The period between the last received message and the end of the MRAI round is called the idle period. During the idle period, the BGP speaker does not process update messages of a particular destination. In this example with 20 update messages per round, durations of the active and idle times are significantly different for the empirical and uniform BGP processing

delays. Their difference may be even more considerable in the case of higher number of update messages per round. The number of update messages received by a BGP speaker may exceed several hundreds per second [31].

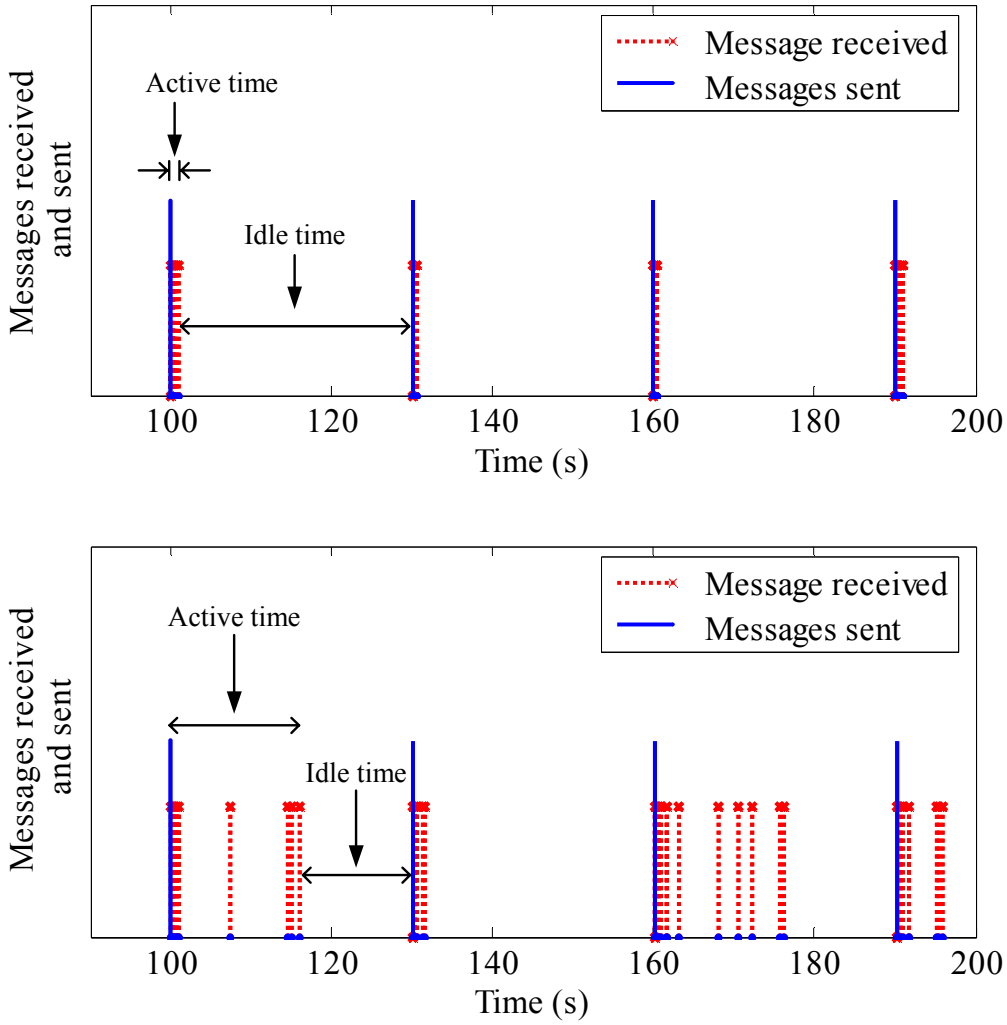

Figure 7. Durations of the active and idle times for the empirical (top) and uniform (bottom) BGP processing delay.

Using the empirical delay value reflects more accurately the behavior of BGP speakers and reveals that they are mostly idle during an MRAI round. These long idle periods increase the BGP convergence time. Hence, minimizing the idle periods optimizes the BGP convergence time. To achieve the optimal BGP convergence time for

one destination, the MRAI value should be chosen close to the active period of each MRAI round.

Using per-destination MRAI timers causes MRAI rounds between BGP speakers close to the origin to start at the same time because the initial update messages reach all BGP speakers at approximately the same time. However, due to different propagation and processing delays of update messages, MRAI timers in the BGP speakers farther from the origin do not start at the same time. As a result, MRAI rounds of two BGP speakers may be shifted in time, as shown in Figure 8.

If MRAI rounds are not synchronised, the idle period cannot be defined as the interval between the last received message and the beginning of the next MRAI round. Instead, we define the idle period as the longest time interval between two received messages in one MRAI round. The active period is then defined as the difference between the duration of an MRAI round and the idle period. The optimal BGP convergence time may be achieved by minimizing the idle period of BGP speakers, as in the case when MRAI rounds start at the same time.

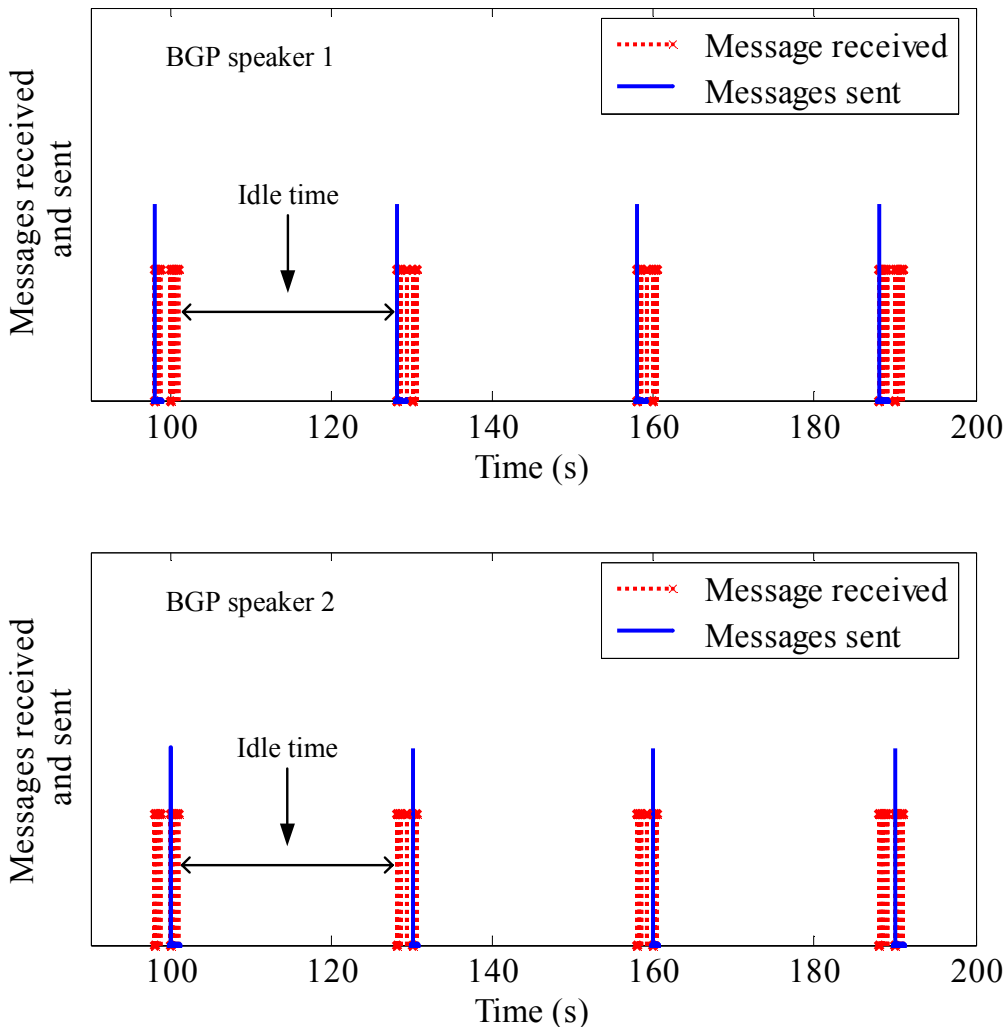

Figure 8. Determining the idle time for two BGP speakers with MRAI rounds starting at different times.

5.2 Reusable MRAI Timers

BGP speakers are active only during a small interval of the convergence time for each destination. Using optimal MRAI values and MRAI timers that perform independent rate limiting for each destination leads to a shorter BGP convergence time. However, per-destination MRAI timers cannot be implemented because of their large number. Per-peer MRAI timers cannot be used because they affect advertisements of all destinations. Therefore, the first step in improving BGP convergence time using the optimal MRAI

values is to design an efficient MRAI timer implementation that independently limits advertisements of various destinations.

The main drawback of per-destination MRAI timers is that they use separate timer for each destination, even when advertisements of these destinations are sent simultaneously. For example, if advertisements of two destinations are sent at time t_1, two per-destination MRAI timers are needed, even though both timers expire at the same time (t_1 + 30 s). Instead of associating one per-destination MRAI timer with each destination, we propose using a single reusable MRAI timer for all route advertisements sent during a certain (short) time interval. We also redefine the rate limiting so that MRAI rounds belong to a certain interval (between 29 s and 30 s), rather than being equal to 30 s. The duration of this interval defines the granularity of the MRAI round that determines the number of reusable MRAI timers. For example, if the granularity is 1 s, a BGP speaker needs 30 reusable timers shifted by 1 s, as shown in Figure 9. The reusable Timer 0 starts at t_0, Timer 1 starts at $t_1 = t_0 + 1$ s, while the last reusable timer (Timer 29) starts at $t_{29} = t_0 + 29$ s. The entire cycle repeats after timer 0 expires at 30 s.

A BGP speaker needs to determine which reusable MRAI timer is to be associated with a sent route advertisement. For each advertisement, the last expired reusable timer is used because it enforces an MRAI round to last between 29 s and 30 s. For example, if the reusable timer 0 starts at 0 s, advertisement 1 sent at time 171.5 s is associated with the reusable timer 21 that starts at 171 s. The duration of MRAI for this advertisement is 29.5 s, as shown in Figure 10. Advertisement 2 sent at 171.7 s is also associated with the timer 21. All other advertisements sent between 171 s and (171 + 1) s are also associated with the timer 21. Reusable MRAI timers enable BGP to handle

advertisements independently, as in the case of per-destination MRAI timers. The number of timers is not greater than the one in the case of per-peer MRAI timers.

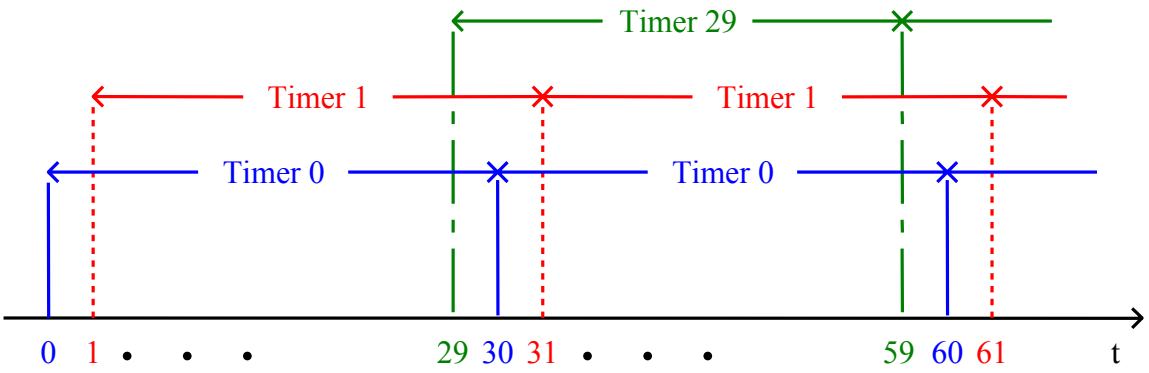

Figure 9. 30 reusable MRAI timers with the granularity of MRAI round is 1 s.

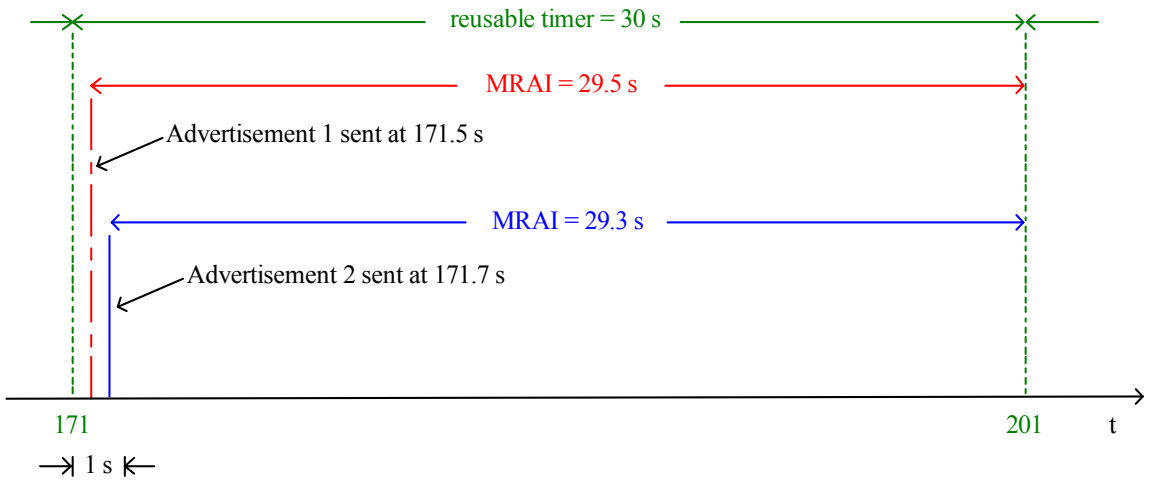

Figure 10. Associating route advertisements with a reusable timer. All advertisements sent between 171 s and (171 + 1) s are associated with the reusable timer 21. Their MRAI round is in the interval between 29 s and 30 s.

The implementation of reusable MRAI timers requires storing pointers that associate each route advertisement with the corresponding timer. Similarly to per-peer MRAI timers, reusable MRAI timers maintain a list of routes that need to be advertised when a timer expires. The overhead for storing pointers to reusable timers is not significant because pointers are needed only for routes that have not converged. A core

Internet router usually has several hundreds non–convergent routes, compared to the entire routing table that may contain over 100,000 routes [31].

5.3 Adaptive MRAI Algorithm

Finding the optimal MRAI value requires knowing the active period during an MRAI round. This active time is not constant and it depends on network conditions. Instead of using a constant global MRAI, we propose the adaptive MRAI algorithm for adjustment of MRAI values. The algorithm is designed to be used with reusable MRAI timers because durations of adaptive MRAI rounds are calculated separately for each destination.

We were unable to use statistical properties to predict the active period because the distributions of the active period durations and inter-arrival times of update messages were not available. Hence, we estimate the active period in the subsequent round using the average active period of the previous rounds. The fluctuations of the active period are estimated using the standard deviation. We also introduce a safety margin to ensure that in the case an active period increases, the subsequent round encompasses all previously sent update messages. The margin is set to three times the standard deviation of an adaptive MRAI round. Thus, the duration of the next adaptive MRAI round for destination D is estimated as:

$$\begin{aligned} &adaptiveMRAI_{n+1}(D) = \\ &avg\_active_n(D) + 3 \times deviation_n(D), \end{aligned} \tag{7}$$

where $avg\_active_n(D)$ and $deviation_n(D)$ are the average duration and the standard deviation of the active period for destination D in the n-th round, respectively. The

minimum duration of the adaptive MRAI round is determined by the number of reusable MRAI timers. For example, for 30 reusable MRAI timers, the minimum duration of the adaptive MRAI round is equal to 1 s.

The adaptive MRAI has *idle* and *processing* states, as shown in Figure 11. The variables are given in Table 3. The algorithm is in the *idle* state for all destinations with stable paths. When a change of destination reachability causes a change of the best route, the BGP speaker sends the first update message to its peers. At that instant, the algorithm enters the *processing* state. This also marks the beginning of the first adaptive MRAI round.

$round_n(D)$	*n*-th round of the BGP convergence process
$adaptiveMRAI_n(D)$	duration of the adaptive MRAI round
$idle_n((D)$	duration of the idle time
$active_n(D)$	duration of the active time
$avg\_active(D)$	the average active time
$deviation_n(D)$	standard deviation of the average active time
$reusable\_timer_n(D)$	serial number of the reusable timer
$last\_received\_update_n(D)$	time when the last update message is received

Table 3. Variables of the adaptive MRAI algorithm used for destination *D* in the n-*th* round.

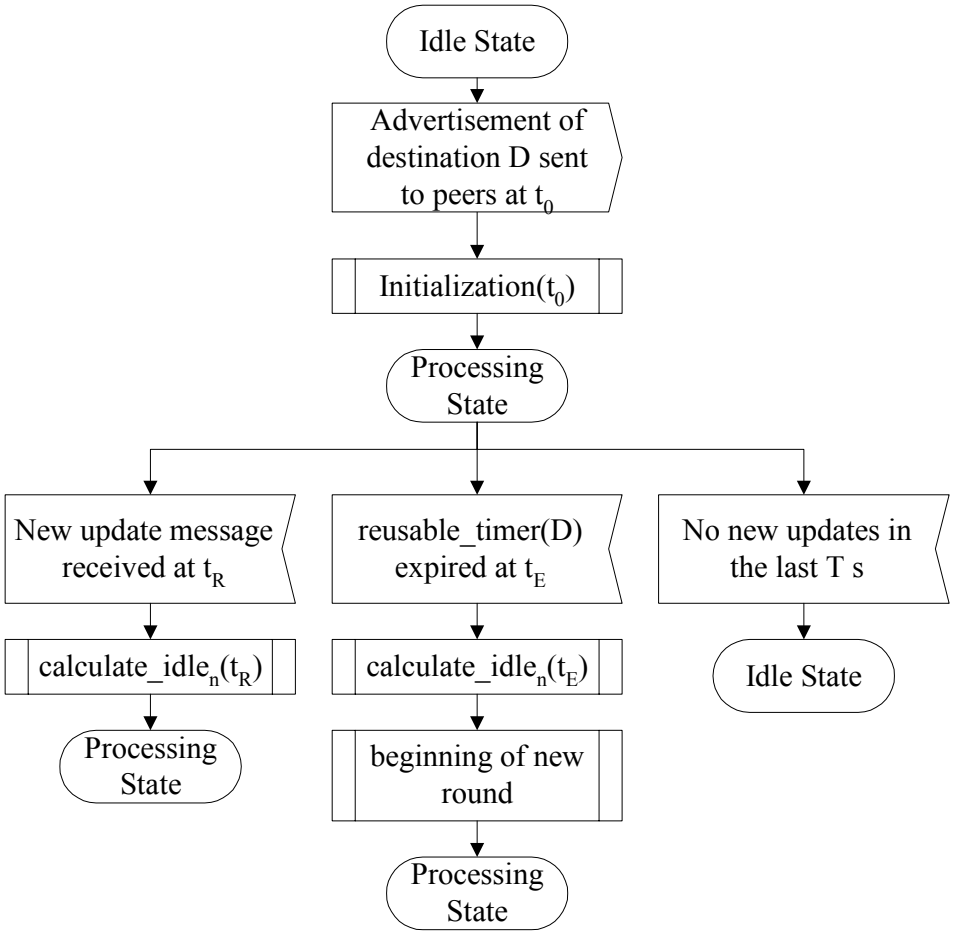

Figure 11. Adaptive MRAI algorithm.

Initial values of the variables are set in the first adaptive MRAI round, as shown in Figure 12. We use the default value of 30 s for the duration of the first round. In addition, we set the standard deviation to 1 s (rather than 0 s) in the first round. Using the standard deviation equal to zero would leave the second adaptive MRAI round without the safety margin in the case of an active time increase.

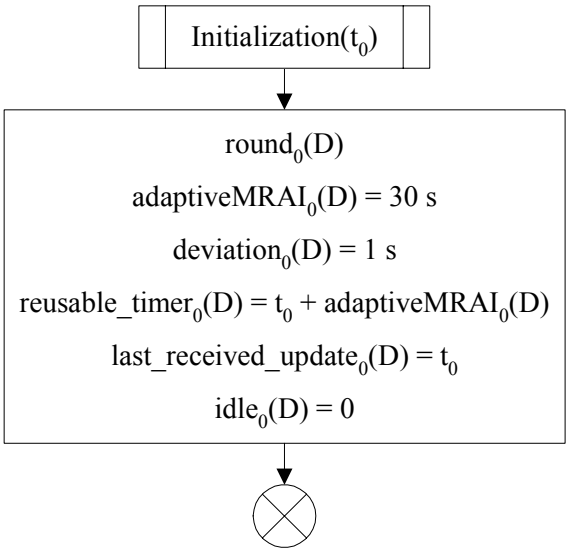

Figure 12. Initialization of the variables in the first adaptive MRAI round.

Three types of events may occur in the *Processing* state: *i*) receiving a new update message, *ii*) expiration of reusable timer associated with a destination, and *iii*) the completion of the BGP convergence process. When a new update message is received, the algorithm remains in the *Processing* state and calculates the idle time, as shown in Figure 13. The idle time is calculated as the longest interval between two update messages in an MRAI round. The expiration of the reusable MRAI timer associated with a destination indicates the beginning of a new adaptive MRAI round. At that time, the BGP speaker recalculates the idle period, the average active period, and the standard deviation of the active period. They are used to predict the duration of the next adaptive MRAI round (7), as shown in Figure 14.

A short idle period during the previous round may indicate that the active period is longer than predicted and that the previous adaptive MRAI round should have lasted longer. The threshold for determining the minimum idle period is set to 1 s, which is identical to the granularity of the adaptive MRAI rounds. If the BGP speaker detects that

the idle period is less than 1 s, it doubles the duration of the next adaptive MRAI round up to 30 s. The maximum duration of adaptive MRAI round (30 s) is equal to the default MRAI value [14]. Hence, the adaptive MRAI algorithm cannot lead to the longer BGP convergence time than the current BGP. At the beginning of a new MRAI round, the BGP speaker assigns again a reusable timer for the destination. Different reusable MRAI timers may be used for the same destination in each adaptive MRAI round.

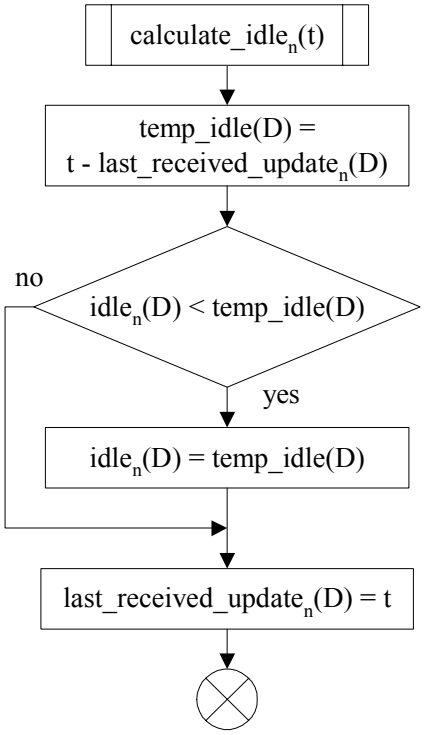

Figure 13. Procedure for calculating the idle time.

We assume that the BGP speaker has converged and that it returns to the *idle* state if it does not receive update messages regarding the destination within a certain previously defined time period *T*.

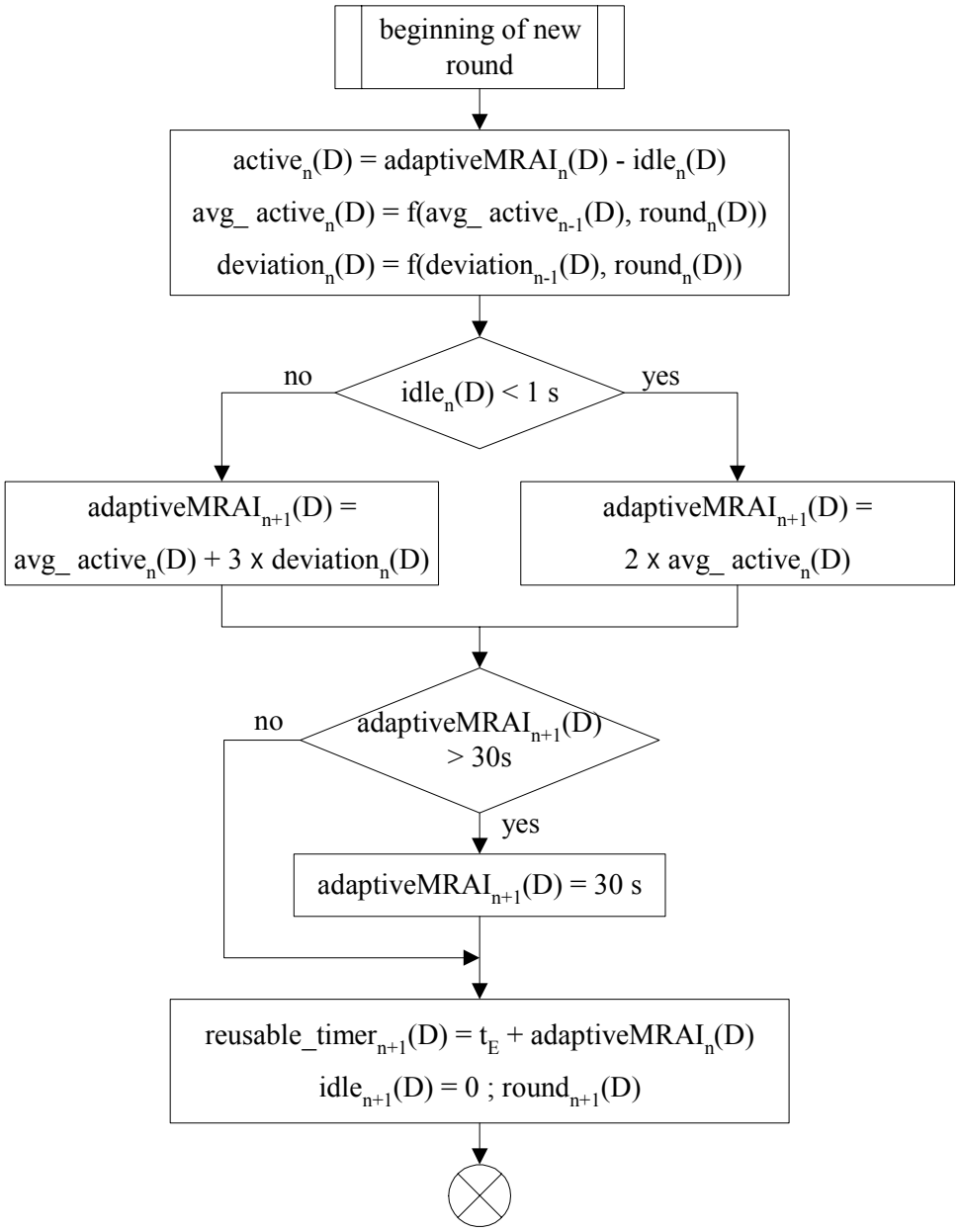

Figure 14. Procedure for recalculating the variables at the beginning of an adaptive MRAI round.

5.3.1 Space and Time Complexity of the Adaptive MRAI algorithm

The details of the BGP implementation in commercial routers are not publicly available. Hence, we address the feasibility of the adaptive MRAI by estimating the

implementation overhead. The adaptive MRAI algorithm depends on the routes that have not converged because the algorithm is used only when the best route to a destination changes. The size of the input n is the number of non-converged routes in a unit of time. This number is approximately several hundreds per second [31]. The overall number of routes in a BGP routing table may be over 100,000 [31].

The implementation of the adaptive MRAI requires the BGP speaker to store four variables for each route non-converged: $round_n(D)$, $avg\_active_n(D)$, $deviation_n(D)$, and $last\_received\_update_n(D)$. $round_n(D)$ is an integer counter, while the remaining three variables contain timestamps in milliseconds. Hence, it is sufficient to use four integers for storage. Other variables listed in Table I may be calculated using these four variables. Therefore, the space complexity of the adaptive MRAI algorithm is $O(n)$ (a linear function of the input size). This overhead per non-converged route is similar to the overhead imposed by the route flap damping algorithm that requires storing three pointers and two integers per flapping route [38].

The time complexity (the running time) of the adaptive MRAI algorithm may be defined as the number of operations performed when the algorithm is in the Processing State (Figure 11). Depending on an event that has occurred, the algorithm in this state may perform one of two following calculations. At the beginning of a new adaptive MRAI round (Figure 14) the algorithm recalculates the variables from Table 3, where it calculates only the idle time for each new received update message (Figure 13). The time complexity of the adaptive MRAI algorithm is a sum of time complexities of these calculations and it usually expressed as a function of the input size n.

The upper bound on the number of adaptive MRAI rounds for each route that has not converged is equal to the granularity of an adaptive MRAI round because the granularity determines the shortest duration of an MRAI round. Thus, there is a linear relationship between the number of adaptive MRAI rounds and the input size n. Further, the time complexity of the calculation at the beginning of a new adaptive MRAI round is a linear function of n. For example, if the granularity is 1 s, then there is at most one adaptive MRAI round per second for each non-converged route.

Operations used in the adaptive MRAI algorithm are: addition, subtraction, multiplication, division, and square root operation. The time complexity of addition and subtraction is constant, where the time complexity of the other three operations depends on the input values. The variables used in operations of multiplication, division, and square root operations are: $active_n(D)$, $avg\_active(D)$, and $deviation_n(D)$. Possible values of these variables are in the range from 0 to 30,000 because the maximum duration of these intervals is 30 s (equivalent to 30,000 ms). To simplify estimation of the time complexity of these three operations, we may approximate the input values with constants equal to their maximum values. This approximation gives the upper bound of the time complexity and enables us to assume that the time complexity of all five used operations does not depend on the input values.

To estimate the number of operations at the beginning of a new adaptive MRAI round, we first estimate the number of operations needed for calculation of the average active time ($avg\_active_n(D)$) and the standard deviation ($deviation_n(D)$).

The average active time in the round n ($avg\_active_n(D)$) is calculated as:

$$
\begin{aligned}
avg\_active_n(D) &= \sum_i^n \frac{active_i(D)}{n} \\
&= avg\_active_{n-1}(D) + \frac{(active_n(D) - avg\_active_{n-1}(D))}{n} \\
&= avg\_active_{n-1}(D) + \frac{\Delta_n}{n}.
\end{aligned}
\tag{8}
$$

where $\Delta_n = active_n(D) - avg\_active_{n-1}(D)$. From (8) it follows that the average active time can be calculated using the value from the previous round ($avg\_active_{n-1}(D)$) and the current value of active time ($active_n(D)$).

The standard deviation of the *active* time ($deviation_n(D)$) is calculated as:

$$
\begin{aligned}
deviation_n(D) &= \sqrt{\sum_i^n \frac{(active_i(D) - avg\_active_n(D))^2}{n}} \\
&= \sqrt{deviation^2_{n-1}(D) + \frac{(\Delta_n^2 - deviation^2_{n-1}(D))}{n}}.
\end{aligned}
\tag{9}
$$

Similar to the calculation of the average *active* time, the standard deviation may be calculated using the value from the previous round ($deviation_{n-1}(D)$) and Δ_n.

From (8) and (9) follows that the maximum number of operations at the beginning of an adaptive MRAI round for each route is constant (4 additions, 3 subtractions, 2 multiplications, 2 divisions, 2 comparisons, and 1 square root operation). Hence, the time complexity of the recalculating the variables at the beginning of an adaptive MRAI round is $O(n)$.

During one MRAI round a BGP speaker sends no more than two update messages (one advertisement and one withdrawal) regarding one route. A BGP speaker cannot send more than one advertisement due to the rate limiting and it cannot withdraw the same

route twice in one MRAI round. Hence, the maximum number of received update messages during one MRAI round for one BGP speaker depends on the number of non-converged routes and the number of its peers. Knowing that the number of peers is a constant for one BGP speaker, the time complexity of the idle time calculation is a linear function of the input size n. As a result, the time complexity of the adaptive MRAI algorithm is a linear function of the input size n, or it is $O(n)$, because both, the calculation of the idle time and the recalculation of the variables are $O(n)$.

Chapter 6

PERFORMANCE OF THE ADAPTIVE MRAI

We implement the adaptive MRAI algorithm, reusable MRAI timers, and the empirical and uniform BGP processing delays in the ns-2 network simulator (ns-2.27) [24]. We use the BGP module for ns-2, ns-BGP 2.0 [9], which had been ported from SSFNET [34].

6.1 ns-2 Implementation

ns-2 is written in both C++ and OTcl (Object oriented Tcl) [25]. We implemented the proposed modifications of BGP using the existing ns-2 and ns-BGP 2.0 C++ class hierarchy. The routing structure of an ns-2 node and the corresponding BGP modules are shown in Figure 15. The address classifier (*classifier_*) determines whether a received packet should be processed in the node or forwarded to another nodes. The port classifier (*dmux_*) forwards packets from the address classifier to the corresponding application, based on their port number. The route object (*rtObject*) is used to coordinate various

dynamic routing protocols in an ns-2 node. The Bellman-Ford's distance vector algorithm is implemented in ns-2 using rtProto/DV agent [25].

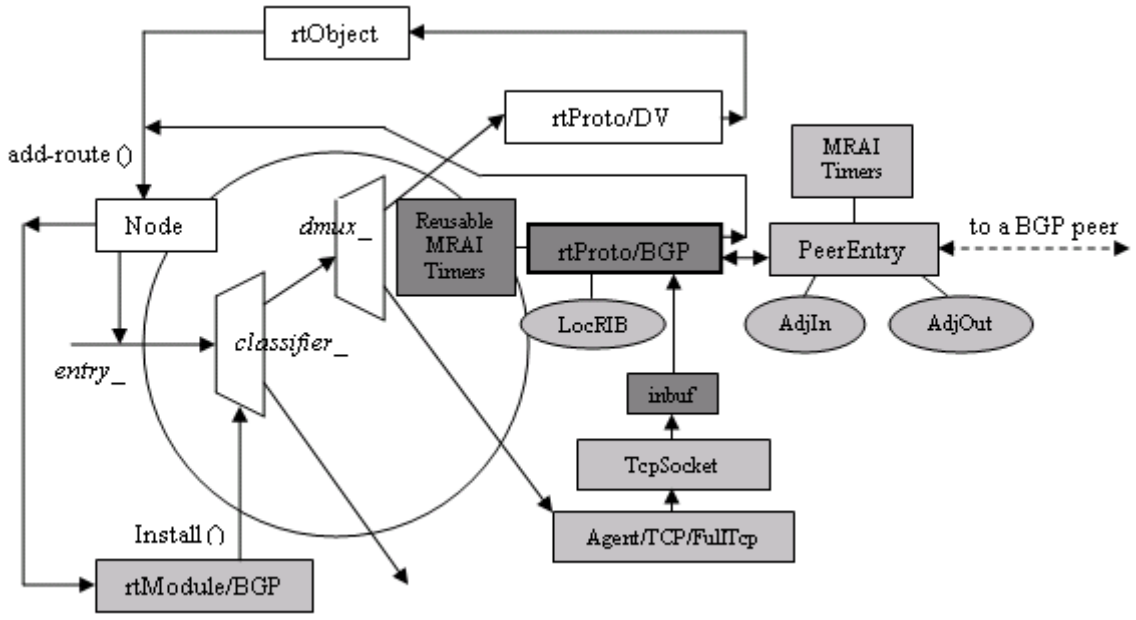

Figure 15. ns-2 routing structure within one node, with added BGP modules (Reusable MRAI timers, rtProto/BGP, and inbuf).

ns-BGP 2.0 consists of the following modules: rtProto/BGP, rtModule/BGP, Agent/TCP/FullTcp, TcpSocket, PeerEntry, MRAI timers, LocRIB, AdjIn, and AdjOut. The original ns-2 routing does not support transfer of user data and IPv4 addressing. To enable them and to achieve compatibility with the SSFNET implementation of BGP, ns-BGP 2.0 adds rtModule/BGP, Agent/TCP/FullTcp, and TcpSocket to the ns-2 routing structure [9]. The C++ class rtProto/BGP performs most BGP operations, such as establishing BGP connections, determining the best route and updating BGP routing table (*LocRIB*), managing the BGP finite state machine [32]. Information regarding each peer connection is stored in the C++ class *PeerEntry*, which contains MRAI timers and two

routing tables: AdjIn (routes learned from a peer) and AdjOut (routes to be advertised to a peer).

To implement BGP-AM and the empirical and BGP processing delays, we modified rtProto/BGP and added reusable MRAI timers and an input buffer (*inbuf*), as shown in Figure 15. BGP-AM uses a single set of reusable MRAI timers associated with the rtProto/BGP module. Conversely, ns-BGP 2.0 uses per-peer MRAI timers that are associated with each PeerEntry. The calculation of adaptive MRAI rounds for each destination is performed in rtProto/BGP. BGP processing delay is modelled using *inbuf*, where BGP update messages are stored before they are processed.

6.2 Simulation Scenarios

We use four network topologies to evaluate the performance of BGP-AM and to compare it with the current implementation of BGP. The simulations were performed on a 2.8 GHz Intel Pentium IV processor with 2 GB of RAM and a Linux Red Hat 9.0 operating system. The average durations of one simulation for the topology with 200 nodes were ~2.5 min and ~2 min, for BGP and BGP-AM, respectively. The Internet contains more than 10,000 ASs [31], yet we decided to limit the size of used topologies to no more than 200 nodes, due to extensive time needed for simulation. For example, the complete set of simulations for the topology with 200 nodes lasted approximately 20 days. Furthermore, the limit that we adopted is similar to limits adopted in the previous studies of BGP convergence time [11], [21], [26], [28] – [30], where the maximum number of nodes in the simulation scenarios does not excide 110.

6.2.1 Simulation Topologies

The first network is a completely connected graph with 15 nodes. Although, this topology does not seem as a realistic representation of the Internet topology, it is used for two reasons. The first reason is that this topology is considered as the worst case scenario for BGP convergence time in the down phase, because it has the maximal number of possible paths for a given number of nodes [16]. The second reason is that ASs at the Tier I level are almost completely connected and may be represented with a completely connected graph [31]. For the same reasons this topology is used in previous studies on BGP convergence time [3], [11], [16], [26].

For the second and third networks, we used topologies originating from the University of Oregon Route Views Project [35]. The main goal of this project is collection and analysis of BGP routing information in the Internet. For that purpose the Route Views Project comprises several BGP routers to capture views of the global Internet routing system from various points in the Internet. Each of the routers has peering sessions with various backbone providers and other ASs. The routers involved in the project do not pass routing information learned from their peers nor forward traffic. Furthermore, they do not advertise any prefixes. The BGP routing tables of the participating routers are stored every 2 hours and are publicly available [35].

As a part of the project seven topologies are derived from collected BGP routing tables [23]. They are derived from the routing table of the Route Views router in AS 6447 (route-views2.oregon-ix.net) on March 18, 2002 at 9:26 AM Pacific Standard Time (Greenwich Mean Time - 8 hours). The part of the used routing table is shown in Table 4. From the routing table it can be seen that the router has several routes for the prefix

3.0.0.0/8 and that it has more than one BGP connection with some of neighbouring ASs (AS 3130).

The Internet topology on the AS level can be derived from connections between BGP speakers. These connections can be deduced from AS paths in BGP routing tables. Two BGP speakers are connected if their AS numbers appear adjacent in an AS path. Fore example, from Table 4 it follows that AS 80 and AS 701 are connected. The topologies obtained from the Route Views routers are too large to be use in simulations and have to be reduced to sizes that are more suitable. The algorithm [23] for reducing the number of nodes merges nodes with the smallest number of peers and prunes links from the merged nodes. The algorithm is applied in several iterations until the topology is reduced to previously chosen size.

Time	Peer's IP	Peer's AS	Prefix	AS path
2002-03-18, 09:26	213.174.64.80	1755	3.0.0.0/8	1755 701 80
2002-03-18, 09:26	147.28.255.2	3130	3.0.0.0/8	3130 7018 80
2002-03-18, 09:26	64.200.199.3	7911	3.0.0.0/8	7911 701 80
2002-03-18, 09:26	147.28.255.1	3130	3.0.0.0/8	3130 7018 80
2002-03-18, 09:26	157.22.9.7	715	3.0.0.0/8	715 1239 80
2002-03-18, 09:26	208.172.146.2	3561	3.0.0.0/8	3561 1239 80
2002-03-18, 09:26	216.18.31.102	6539	3.0.0.0/8	6539 701 80
2002-03-18, 09:26	199.74.221.1	8121	3.0.0.0/8	8121 6461 701 80

Table 4. An example of BGP routing table updates used for generating simulations topologies.

The fourth topology has 200 nodes and it is obtained using topology generator BRITE [22]. It supports several models for generating Internet topologies [2], [39]. These models are designed to accurately capture the Internet topology on various levels, such as

ASs' topologies, routers' level topologies, and LANs' topologies. One of the most important findings regarding the Internet topology is that some topological properties can be described using power-laws [5]. For example, the number of nodes with the same number of neighbours versus the number of node's neighbours is a power-law function [5]. Therefore, for a given set of nodes the choice of links among them has to result in a power-law distribution of the nodes' neighbours.

To generate the fourth topology we used Barabási-Albert's model of AS level topologies [2]. The Barabási-Albert's model incrementally adds new nodes to a network, where the probability that a new node will connect to an old node is proportional to the number of the old nodes neighbours. Consequently, new nodes tend to connect to the old nodes with the higher number of neighbours and the topology has power-law distribution of nodes' neighbours.

6.2.2 Simplifications Adopted in Simulations Scenarios

In order to observe only the influence of the adaptive MRAI algorithm on BGP convergence time we choose to adopt several simplifications. First, we do not consider scenarios with long-term instabilities (route flaps). These instabilities lead to long BGP convergence time and may mask affects of the adaptive MRAI algorithm, because a route may be suppressed significantly longer than the duration of an MRAI round. Second, we considered only cases when the routing policies are not applied, because they may cause persistent route oscillations [36] without regard on the implementation of MRAI. Third, to limit the number of BGP speakers in simulations, we assume that one AS is represented with a single BGP speaker. A result of this assumption is that the influence on the iBGP (Interior BGP) on BGP convergence time is not considered.

In each simulation is considered only one prefix, because the adaptive MRAI algorithm treats independently update messages for each destination. In addition, the impact of all other update messages on the average BGP processing delay is modeled by the empirical BGP processing delay. Each simulation scenario is repeated 30 times using 30 unique random number generator (RNG) seeds. Shown are the average values for each simulation scenario. The RNG is used to randomly shift the starting times of reusable MRAI timers, per-peer MRAI timers, and processing cycles of update messages in distinct BGP speakers. We assume that per-peer MRAI timers are working continuously, which mimics the observed behavior of BGP speakers [17]. BGP speakers use the sender side loop detection (SSLD) mechanism and limit only the advertisements (withdrawal limiting is not used). We also assume that the BGP convergence process is completed if no update message is exchanged between BGP speakers within T equal to 60 s. If not stated otherwise, the adaptive MRAI employs 30 reusable MRAI timers (with granularity of 1 s) and a 200 ms processing cycle for the empirical BGP processing delay.

6.3 Completely Connected Graph with 15 Nodes

The completely connected graph with 15 nodes is shown in Figure 16. The choice of the origin in a completely connected graph does not affect simulation results because of the graph symmetry. The *up* phase is not simulated because all nodes are directly connected to the origin and, hence, BGP converges almost instantaneously.

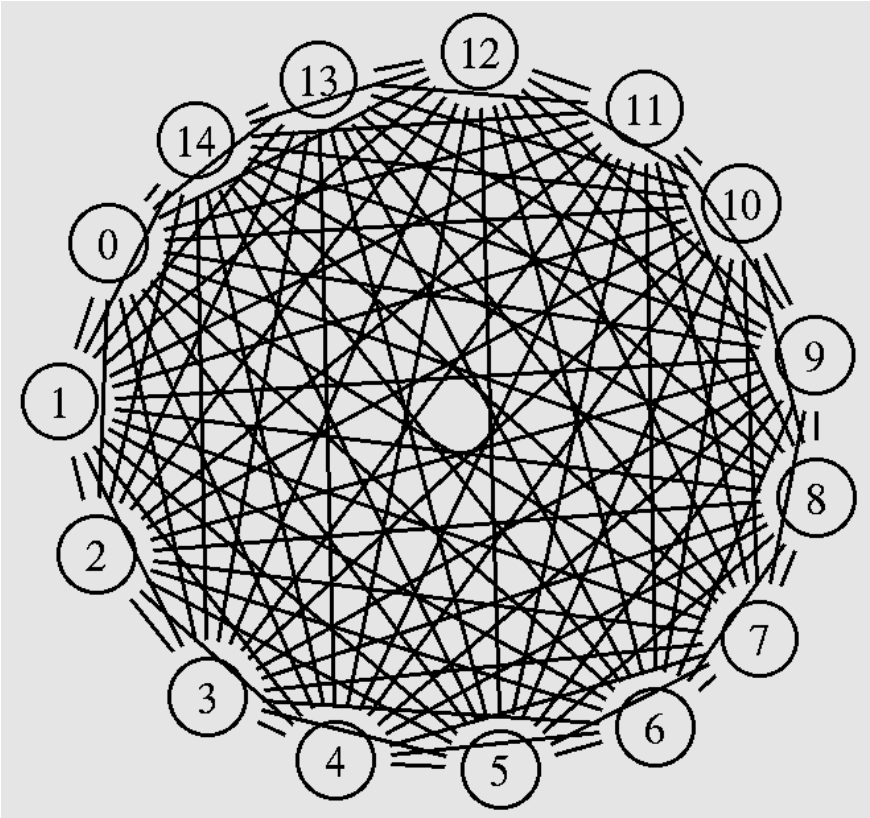

Figure 16. Completely connected graph with 15 nodes.

The relationship between BGP convergence time and the duration of the MRAI round for the empirical and uniform BGP processing delays in the down phase is shown in Figure 17 (top). The results when the uniform delay is used are similar to the results reported in [11] and [26]. Slight discrepancies may be caused by different simulation scenarios. For example, in [11] BGP speakers do not use SSLD and continuous per-peer

MRAI timers. A relationship between the overall number of update messages exchanged during the BGP convergence process and MRAI is shown in Figure 17 (bottom).

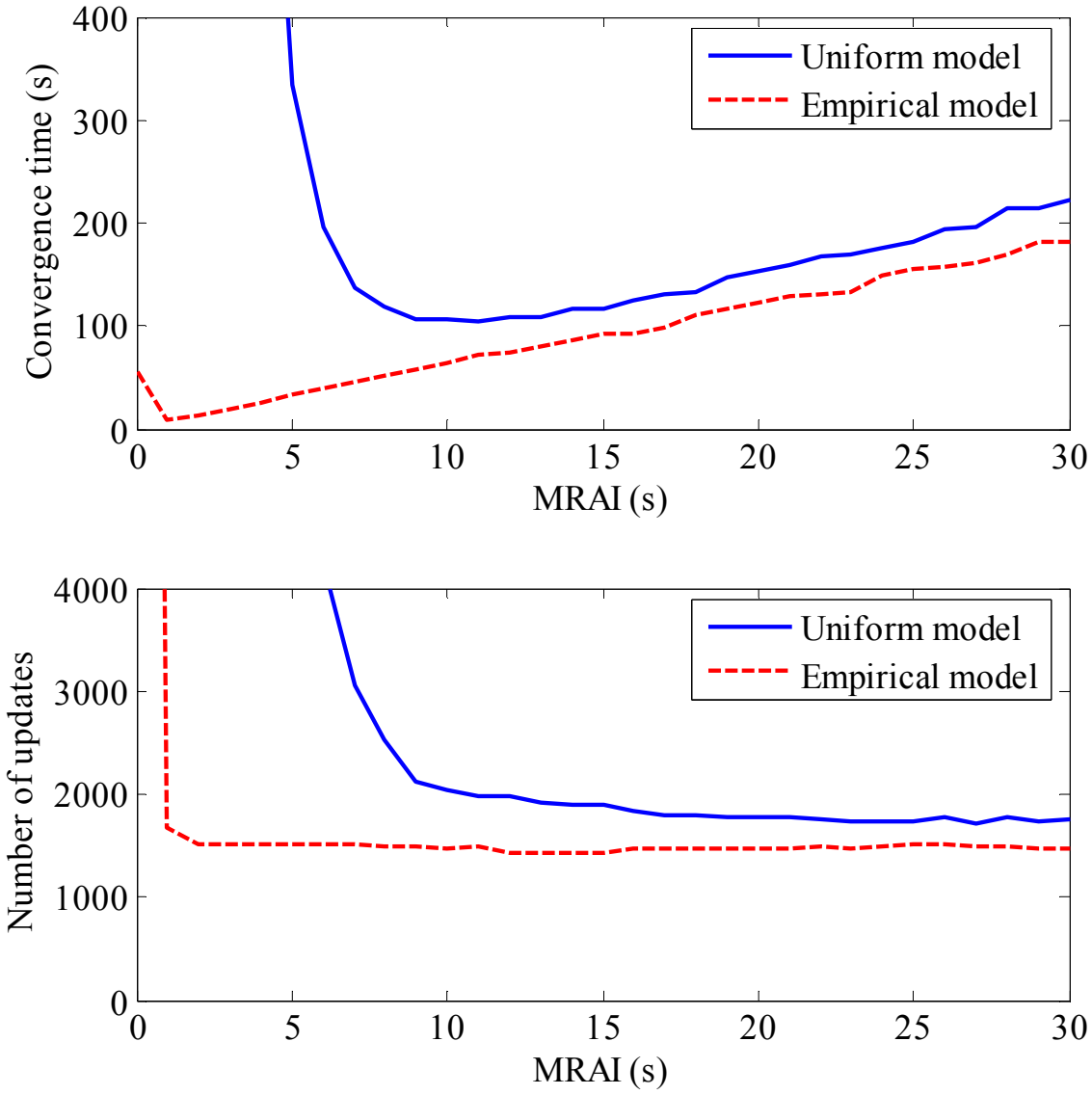

Figure 17. Down phase: BGP convergence time (top) and the number of update messages (bottom) vs. the duration of MRAI.

We define the optimal MRAI value M_o as the value of MRAI that minimizes the duration of BGP convergence time, as is shown in Figure 18, which is a zoom-in of Figure 17 (top). Using M_o does not necessarily result in the minimal number of update

messages. The optimal MRAI value may be viewed as the minimal time needed for processing all update messages received in one MRAI round. Furthermore, use of the optimal MRAI minimizes the idle time of the BGP speakers.

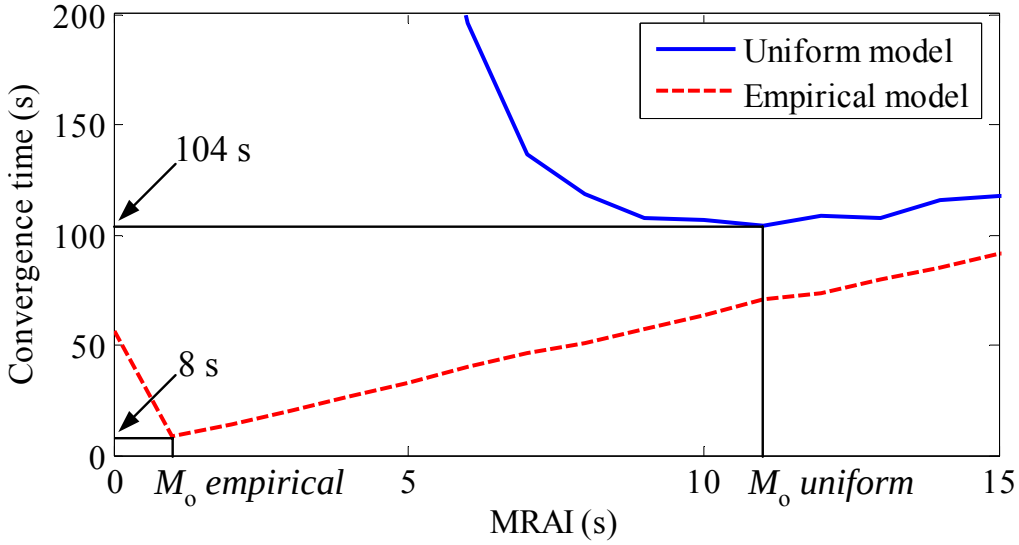

Figure 18. Optimal values of MRAI for the uniform and empirical BGP processing delay.

The optimal MRAI value for the empirical delay (1 s) is shorter than for the uniform delay (~11 s), as shown in Figure 18. The reason for this difference of the optimal MRAI values is the difference of the average processing delays, that are 200 ms and ~7 s (6) for the empirical and uniform delay, respectively. A decrease in the average processing delay implies that BGP speakers need less time to process update messages, which leads to shorter M_o. The minimum BGP convergence times for the empirical and uniform delays are 8 s and 104 s, respectively.

Using MRAI values greater than M_o prolongs the idle time of BGP speakers and, thus, increases BGP convergence time. Hence, BGP convergence time is a linear function of MRAI for values greater than the optimal, as shown in Figure 17. This same linear

relationship between BGP convergence time and the duration of MRAI round would be obtained if the theoretical upper bound for BGP convergence [17] is used (5). Therefore, our simulation results agree with both previous simulations [11], [26] and theoretical analysis of BGP convergence time [17]. BGP convergence time with the uniform BGP processing delay increases significantly for MRAI values smaller than the optimal. BGP speakers in this case cannot process all the received updates and have to make decisions without complete knowledge of their peers' status. For the empirical BGP processing delay, BGP convergence time is small even when MRAI is 0 s because of the much shorter average processing delay. Furthermore, the processing cycle of 200 ms in the empirical BGP processing delay is also acting as a rate limiting factor, similar to an MRAI timer. Even when the rate limiting is not implemented (MRAI = 0 s), a BGP speaker cannot respond to updates instantaneously because it has to wait until the end of the 200 ms processing cycle.

The overall number of update messages for both uniform and empirical BGP processing delays is similar and constant for MRAI values larger than the optimal, as shown in Figure 17 (bottom). That is to be expected, because an increase of MRAI only prolongs the idle period, while the number of exchanged messages remains identical. This suggests that there is the minimum number of messages that BGP speakers have to exchange in order to complete the BGP convergence process. However, for MRAI values smaller than the optimal, the overall number of messages grows significantly, which burdens all BGP speakers in the network.

6.3.1 Performance of the Adaptive MRAI Algorithm

The average BGP convergence time for BGP with the default value of MRAI = 30 s is 181.3 s, as illustrated in Figure 19. The adaptive MRAI algorithm decreases it to 45.1 s. The average number of update messages is similar: 1,480 and 1,553 for BGP and BGP-AM, respectively. BGP convergence time and the number of update messages are not a function of MRAI when the adaptive MRAI algorithm is used. We plot them to illustrate the difference between BGP and BGP-AM.

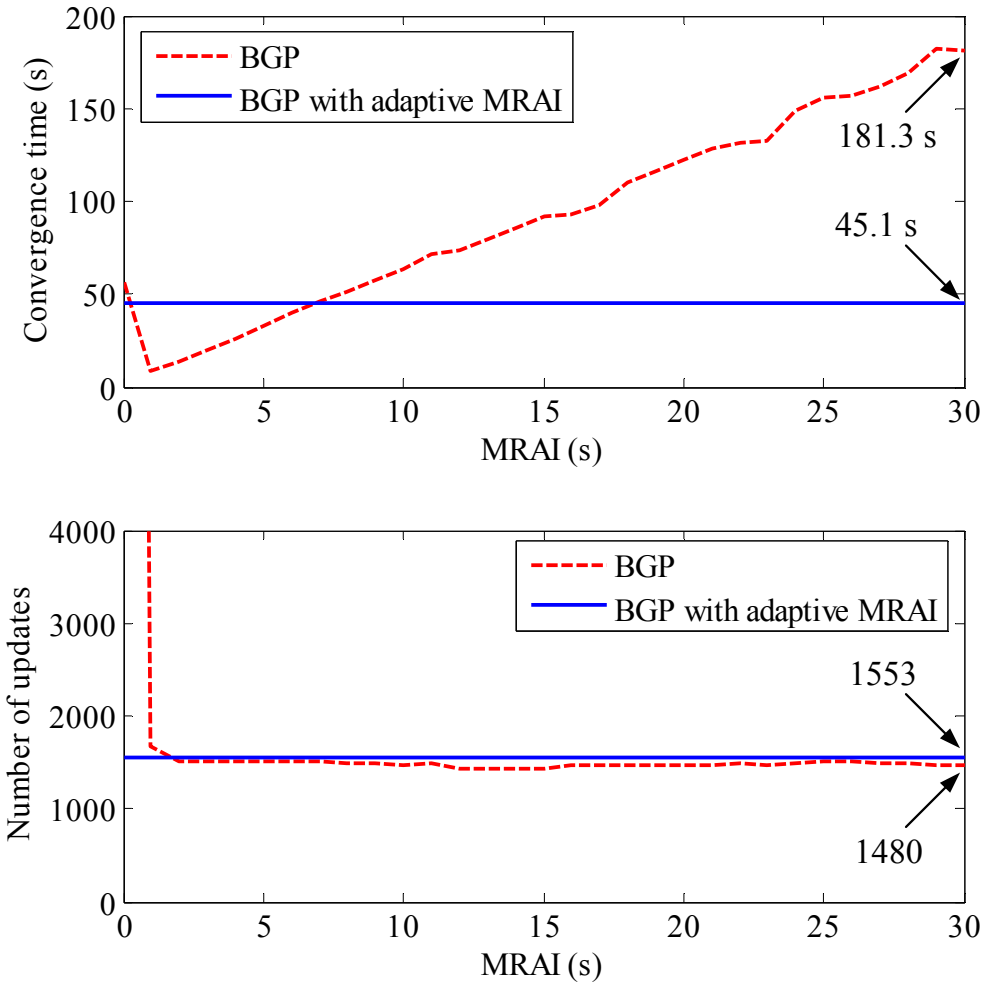

Figure 19. Down phase: Comparison of BGP and BGP-AM. Convergence time (top) and the number of update messages (bottom) vs. the duration of MRAI.

The exchange of update messages during the BGP convergence process in the down phase is shown in Figure 20. The adaptive MRAI adjusts durations of MRAI rounds to minimize the idle time, as shown in Figure 20 (top). On the contrary, durations of an MRAI round are constant when per-peer MRAI timers are used, as shown in Figure 20 (bottom). The overall number of MRAI rounds is identical in both cases. The shorter BGP convergence time for the adaptive MRAI is a result of optimizing the duration of the MRAI rounds.

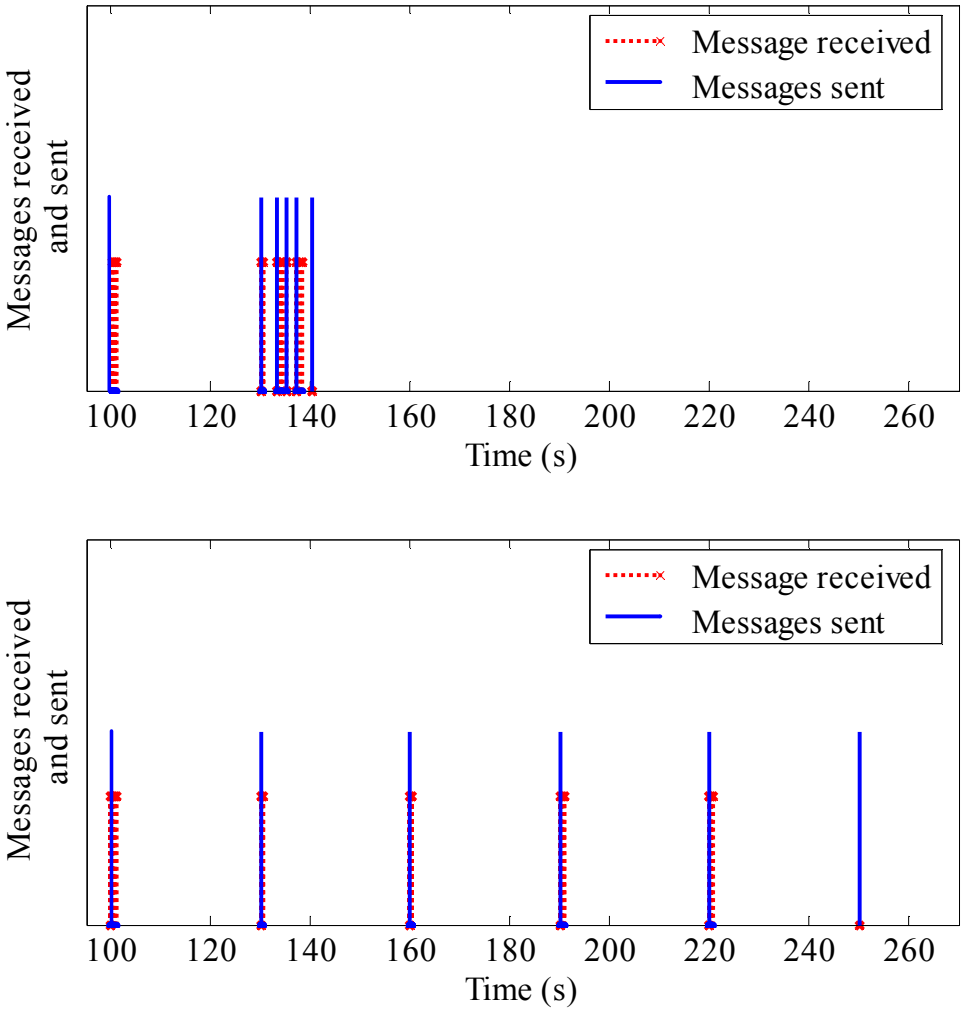

Figure 20. Durations of MRAI rounds for BGP-AM (top) and BGP (bottom).

BGP speakers are mostly idle during the first adaptive MRAI round (equal to 30 s). Hence, the adaptive MRAI does not achieve the minimal BGP convergence time (8 s) as in the case when the optimal MRAI value (1 s) is used, (shown in Figure 19 (top)). Using the first adaptive MRAI round shorter than 30 s results in a shorter BGP convergence time. For example, if the duration of the first adaptive MRAI round is equal to 5 s, the average BGP convergence time is equal to 25.1 s. We set the first adaptive MRAI round to 30 s in order to ensure that BGP speakers may estimate longer idle times (up to 30 s). Note that the optimal value for the first adaptive MRAI round may be best found through measurements.

BGP convergence time and the number of reusable MRAI timers for the down phase are given in Table 5. The number of reusable MRAI timers determines the granularity of MRAI and the minimal duration of the adaptive MRAI round. Although a finer granularity is desired, decreasing the granularity implies increasing the number of timers and the complexity of the implementation. Therefore, the number of MRAI timers should be a trade-off between complexity and performance.

Reusable timers		Convergence time (sec)	Number of updates
Number of timers	Granularity (sec)		
10	3	56.2	1651.7
15	2	54.9	1659.8
30	1	45.1	1552.9
60	0.5	45.8	1489.0
120	0.25	45.7	1520.0

Table 5. BGP convergence time and number of update messages for various number of reusable MRAI timers.

Table 5 indicates that for granularities finer than or equal to 1 s, BGP convergence times are similar. They increase for granularities coarser than 1 s. Using the coarser granularities prolongs the idle time because the maximal BGP processing delay is only 200 ms. The optimal granularity is 1 s. It achieves similar performance with fewer timers than for granularities of 0.5 s and 0.25 s.

The relationship between BGP convergence time and the duration of the BGP processing cycle in the case of the empirical BGP processing delay is given in Figure 21 (top). The minimal value of the processing cycle of 200 ms corresponds to the normal state of a BGP speaker operation. The maximal value of 3 s corresponds to situations when BGP speakers encounter high traffic loads.

The average BGP convergence time (dashed line) does not depend significantly on the duration of processing cycles. This is to be expected because the default duration of MRAI provides sufficient time for BGP speakers to process all received messages. When the adaptive MRAI is used (solid line), the average BGP convergence time is a linear function of the processing cycle because the adaptive MRAI adjusts the duration of MRAI rounds based on the average processing delay. Hence, the increase of the average processing delay results in a proportional increase of BGP convergence time. The overall number of update messages is similar for both cases (~1500), as shown in Figure 21 (bottom). This is consistent to the values shown in Figure 19 (bottom). The duration of the BGP processing cycle does not significantly affect the overall number of update messages.

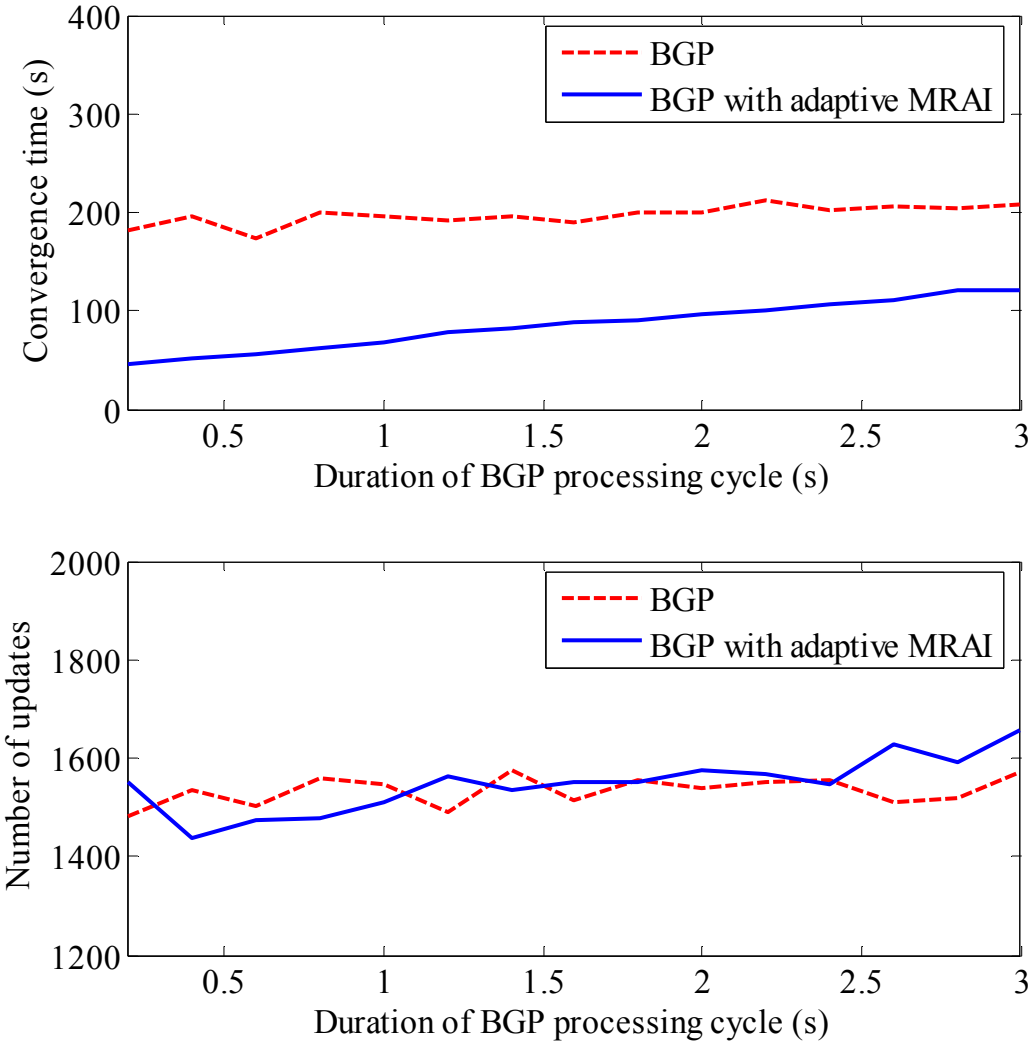

Figure 21. Down phase: BGP convergence time (top) and the number of update messages (bottom) vs. the duration of the BGP processing cycle. (The y-axis on the bottom figure starts from 1200.)

6.4 Network with 29 Nodes

Simulation results for the up and down phases for the network with 29 nodes (Figure 22) are given in Figure 23 and Figure 24, respectively. The results are obtained using the empirical BGP processing delay for both BGP (dashed lines) and BGP-AM (solid lines).

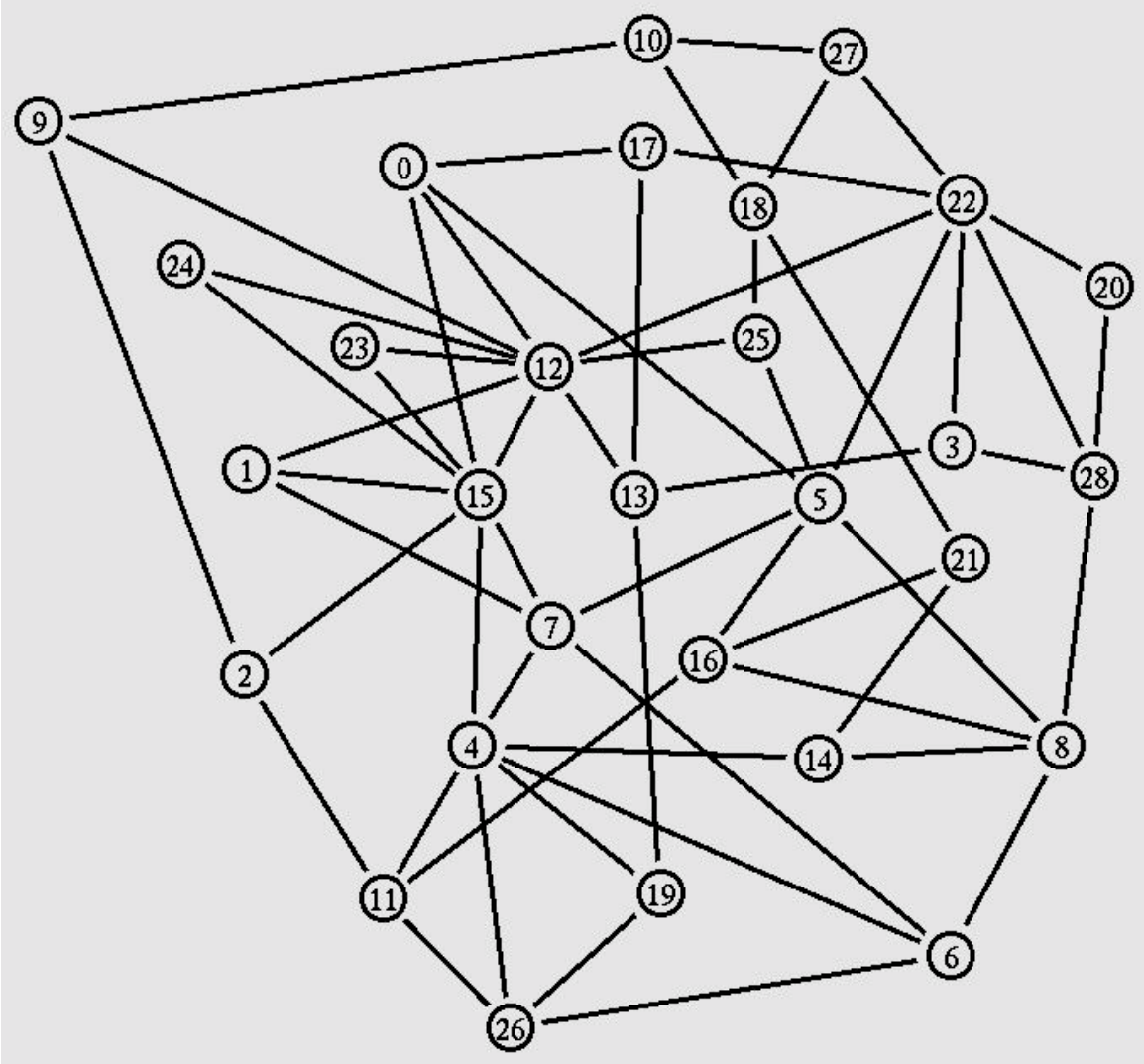

Figure 22. Network with 29 nodes.

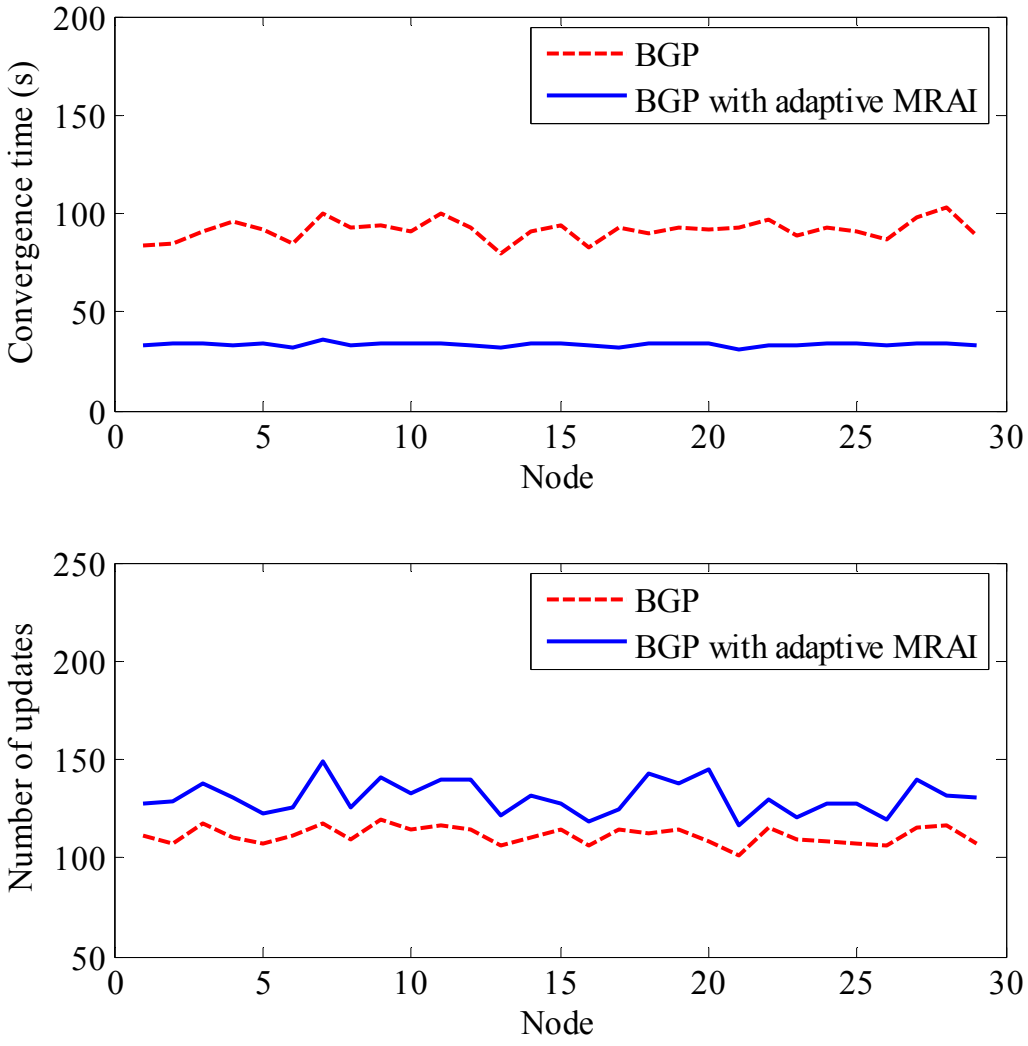

Figure 23. Up phase: BGP convergence time (top) and the number of update messages (bottom) for various origins (nodes) for the network with 29 nodes. (The y-axis on the bottom figure starts from 50.)

BGP convergence time depends on the length of paths from the origin to other BGP speakers. Hence, we repeat simulations using every node in the network as the origin. The simulation results show that BGP convergence time and the overall number of messages vary with the choice of the origin. Using BGP-AM algorithm results in smaller durations and smaller variations in BGP convergence time in both up and down phases.

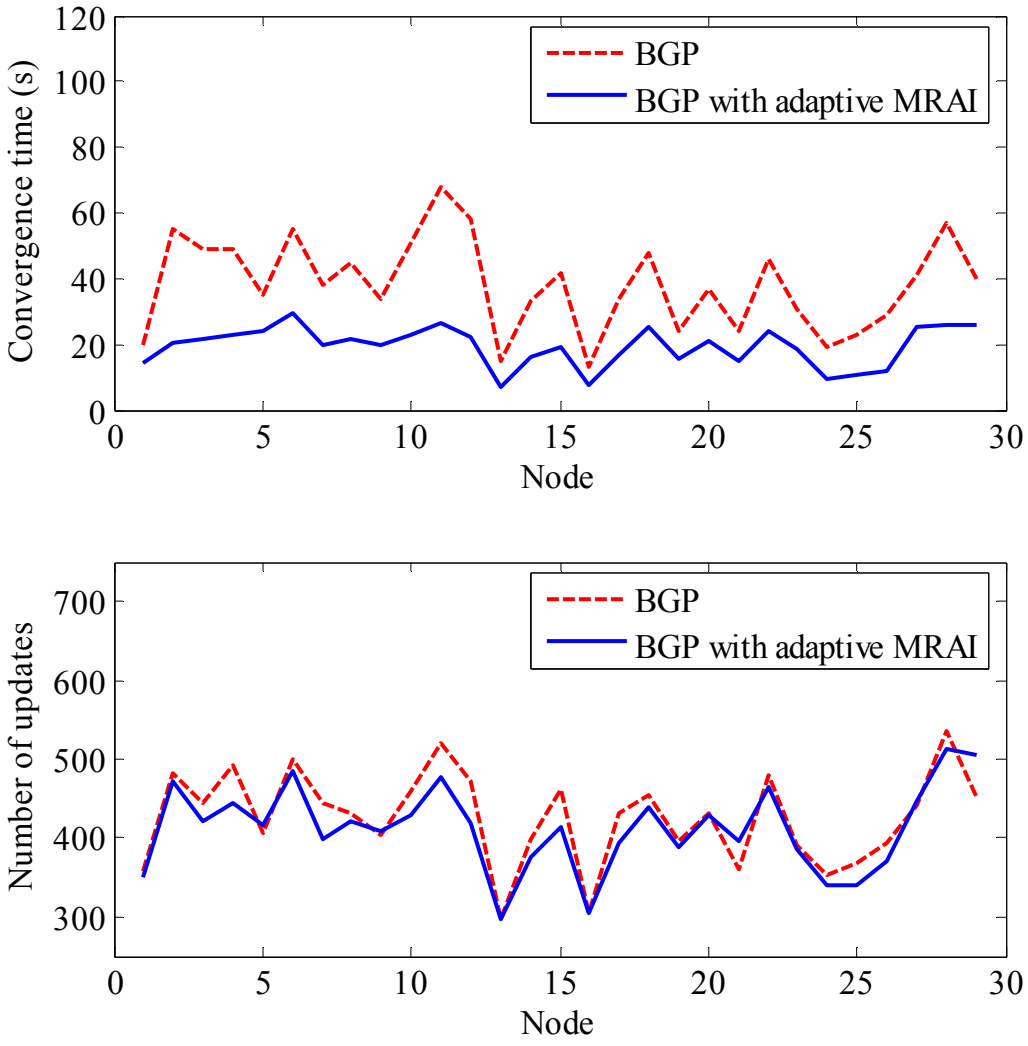

Figure 24. Down phase: BGP convergence time (top) and the number of update messages (bottom) for various origins (nodes) for the network with 29 nodes. (The y-axis on the bottom figure starts from 250.)

In the up phase, without the adaptive MRAI, the majority of BGP speakers needs 3 MRAI rounds to learn the best route, resulting in an average BGP convergence time of ~90 s, as shown if Figure 23 (top). However, with the adaptive MRAI, all BGP speakers in the up phase learn the best route to the destination after 3 s. However, some of BGP speakers first send update messages for non-optimal routes. They have to wait until the

end of the first MRAI round to send new updates. Although, these updates do not change the best routes, they affect BGP convergence time (~30 s), as shown in Figure 23 (top).

Due to the small diameter of the network, in the down phase BGP converges in less than 3 s in ~30% of simulations regardless of the implementation of MRAI timers, as shown in Figure 24. Using the adaptive MRAI results in shorter BGP convergence time when the BGP convergence process lasts more than one MRAI round.

The overall number of messages is similar for both BGP and BGP-AM in the down phase. In the up phase, the adaptive MRAI causes ~10% additional messages.

6.5 Network with 110 Nodes

Simulation results for up and down phases for the network with 110 nodes are shown in Figure 25 and Figure 26, respectively. We repeat simulations using every node as the origin.

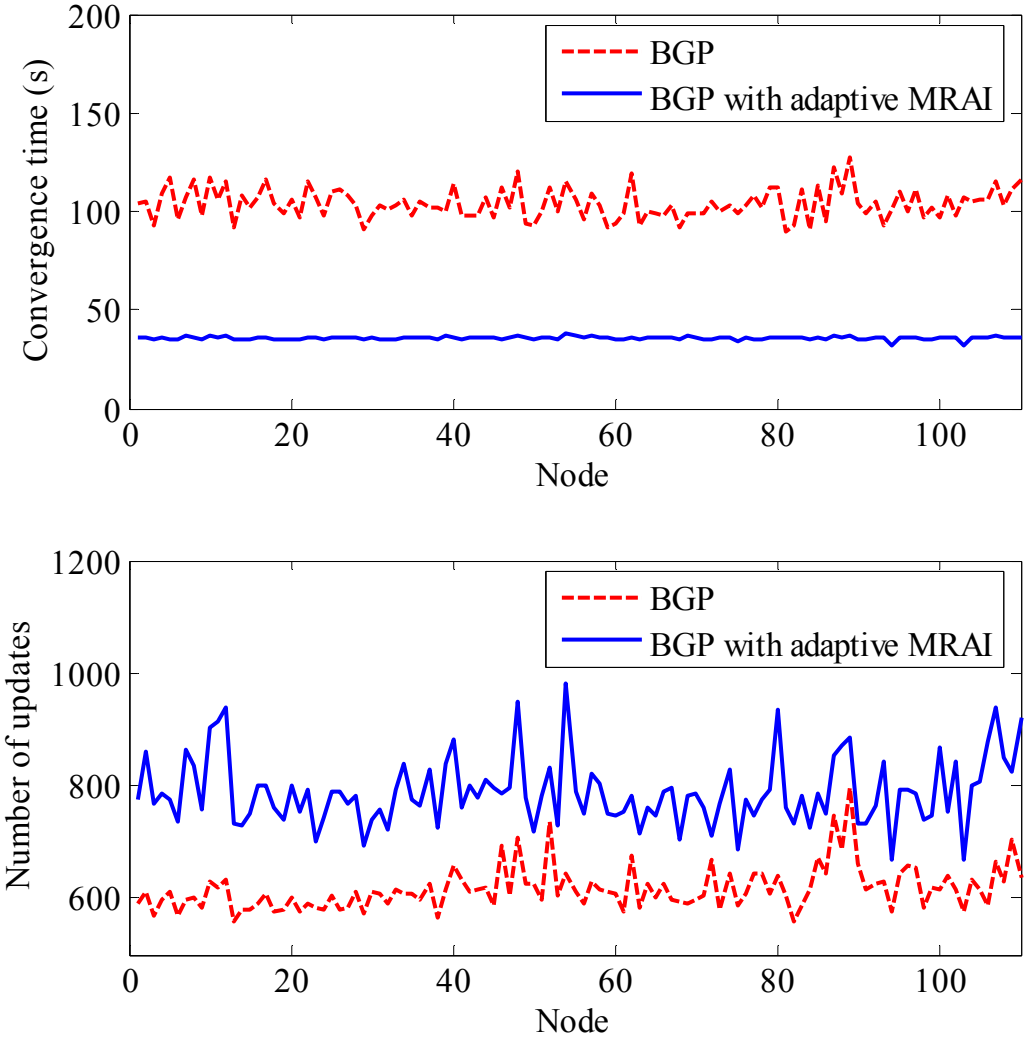

Figure 25. Up phase: BGP convergence time (top) and the number of update messages (bottom) for various origins (nodes) for the network with 110 nodes. (The y-axis on the bottom figure starts from 500.)

The BGP convergence processes last ~4 MRAI rounds for the up phase and ~20 MRAI rounds for the down phase, due to the large network diameter. Using the adaptive

MRAI decreases BGP convergence times from ~120 s to ~35 s in the up phase (Figure 25) and from ~600 s to ~100 s in the down phase (Figure 26). The adaptive MRAI increases the overall number of update messages in the up phase by ~30% and decreases it in the down phase by ~20%.

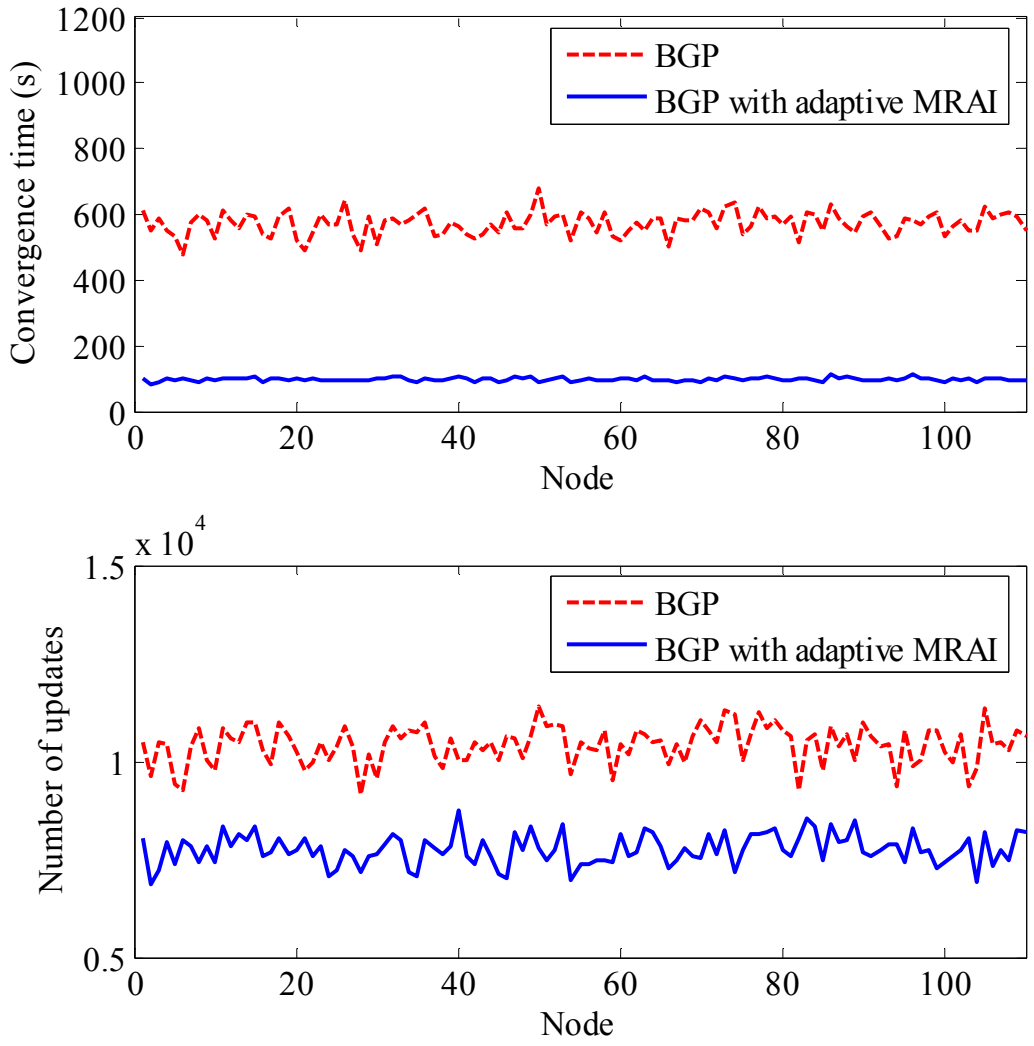

Figure 26. Down phase: BGP convergence time (top) and the number of update messages (bottom) for various origins (nodes) for the network with 110 nodes. (The y-axis on the bottom figure starts from 5,000.)

6.6 Network with 200 Nodes Generated Using BRITE

Using the topology generator BRITE [22] we generated the network topology with 200 nodes. Simulation results for up and down phases are shown in Figure 27 and Figure 28, respectively. Similarly to previous two topologies we repeat simulations using every node as the origin of the topology change.

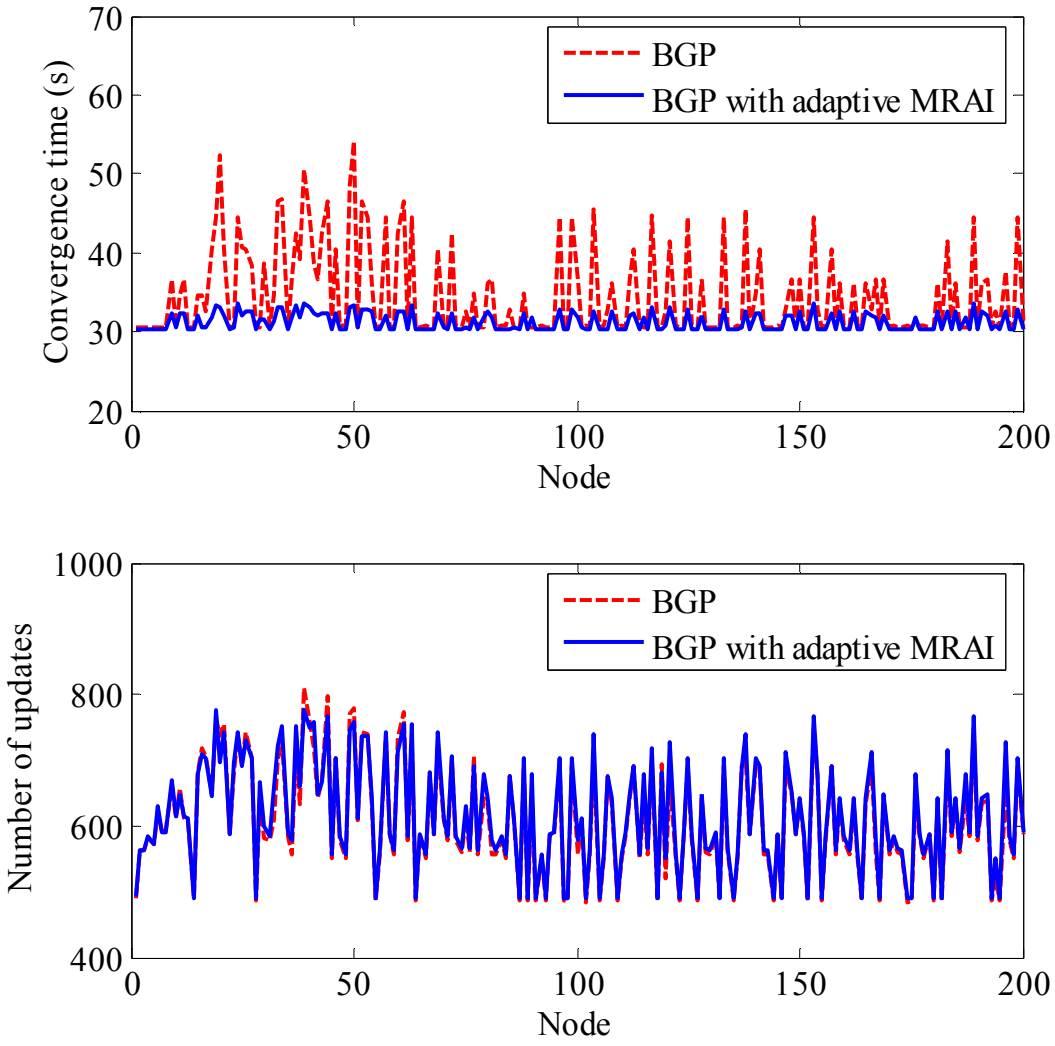

Figure 27. Up phase: BGP convergence time (top) and the number of update messages (bottom) for various origins (nodes) for the network with 200 nodes. (The y-axis on the bottom figure starts from 400.)

Using the adaptive MRAI decreases BGP convergence times in the up phase for

simulations scenarios that lasted more than one MRAI round, while the overall number of update messages is almost identical for both BGP and BGP-AM. In the down phase using the adaptive MRAI algorithm decreases BGP convergence time from ~300 s to ~60 s and the overall number of update messages by ~10%.

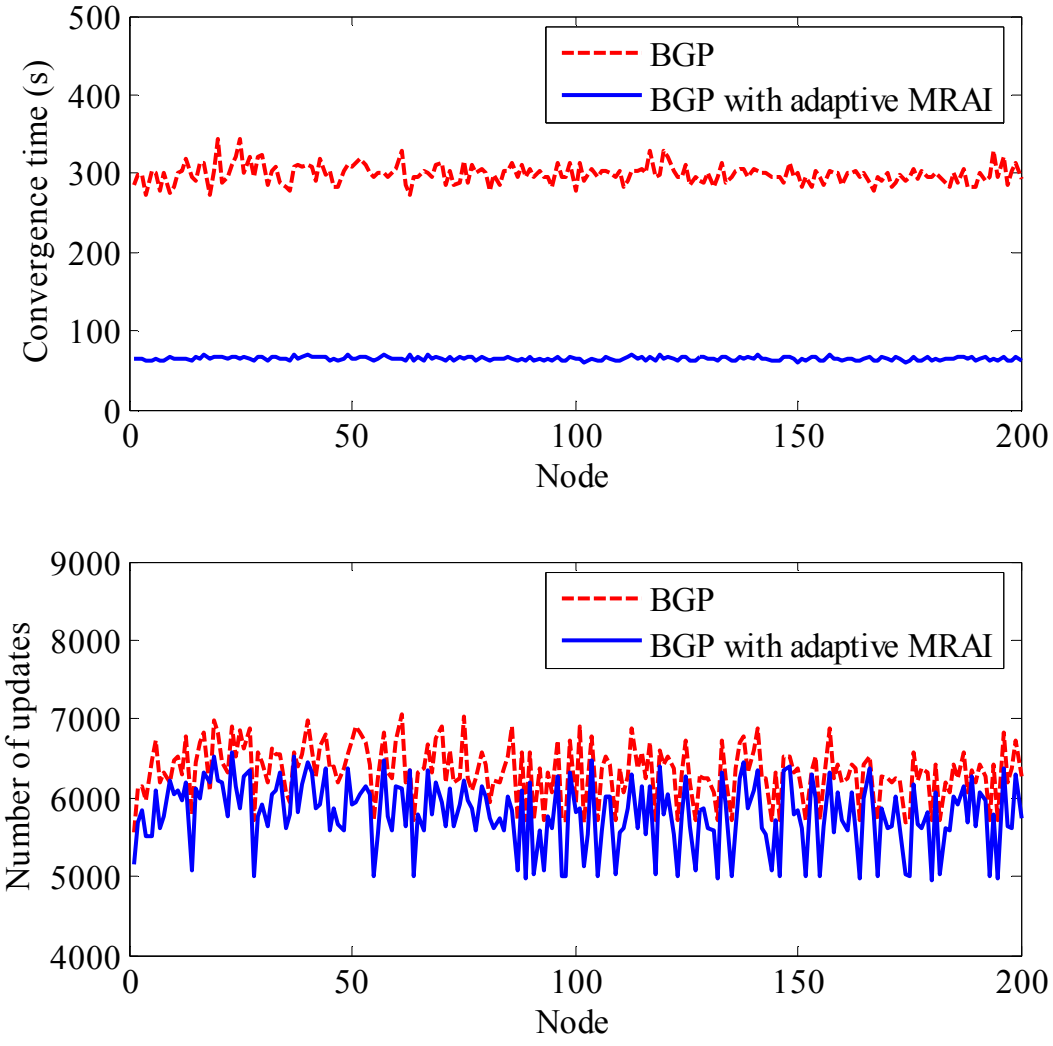

Figure 28. Down phase: BGP convergence time (top) and the number of update messages (bottom) for various origins (nodes) for the network with 200 nodes. (The y-axis on the bottom figure starts from 4,000.)

Chapter 7

CONCLUSIONS

In this thesis, we presented the adaptive MRAI algorithm that enables BGP speakers to adjust the duration of MRAI rounds based on the number of received update messages. The algorithm employs reusable MRAI timers. We also introduced an empirical BGP processing delay, a new approach for estimation of the BGP processing delay based on measurements [1], [7].

We implemented the proposed BGP modifications in ns-2. The simulation results show that adaptive MRAI leads to shorter BGP convergence times in both up and down phases for four simulated network topologies, while the overall number of exchanged update messages remained similar to the current BGP implementation. With adaptive MRAI, BGP convergence time is a linear function of the average BGP processing delay. As in the case of BGP, it depends on the number of MRAI rounds needed for BGP

convergence. Faster BGP convergence is particularly important in the case of larger networks when the BGP convergence process lasts several MRAI rounds.

Further improvement of the adaptive MRAI algorithm requires additional measurement of the processing delay, the active time durations, and update messages inter-arrival times in various network settings. These measurements may suggest different values for the algorithm's variables and parameters. For example, the duration of the next adaptive MRAI round may be more accurately predicted from the distribution of the active time durations, instead using the average value of the active times. Similarly, the safety margin may be calculated using the deviation of the predicted value.

Simulations results indicate that BGP speakers are idle most of the first MRAI round and that decreasing the duration of the first round results in shorter BGP convergence time. Measurements in deployed networks are needed for finding the optimal value of the first MRAI round. Furthermore, in the case of slow fluctuations of the average BGP processing delay, the algorithm may be simplified because the duration of the adaptive MRAI may not need to be recalculated in each round.

Additional improvements of the BGP convergence process may be achieved by using the adaptive MRAI algorithm in conjunction with one of the algorithms designed to eliminate invalid routes [2], [27], [29], described in Section 4.1. These algorithms decrease the number of MRAI rounds during the BGP convergence process, while they do not affect the duration of MRAI rounds. Therefore, incorporating the adaptive MRAI algorithm with one of the algorithms described in Section 4.1 could lead to shorter BGP convergence time than in the case when each of algorithms is used independently.

Future work may also include simulations with settings different from those presented in this thesis. For example, the simulations could be repeated using both BGP speakers that support the adaptive MRAI algorithm and BGP speakers that support only the current implementation of BGP. Furthermore, the algorithm should be tested in scenarios when route flaps occur in order to investigate its interaction with route flap damping mechanisms.

Bibliography

[1] S. Agarwal, C. Chuah, S. Bhattacharyya, and C. Diot, "Impact of BGP dynamics on router CPU utilization," in *Proc. PAM*, Antibes Juan-les-Pins, France, Apr. 2004, pp. 278–288.

[2] A. L. Barábasi and R. Albert, "Emergence of scaling in random networks," *Science*, Oct. 1999, pp. 509–512.

[3] A. Bremler-Barr, Y. Afek, and S. Schwarz, "Improved BGP convergence via ghost flushing," in *Proc. INFOCOM*, San Francisco, CA, March–Apr. 2003, pp. 927–937.

[4] J. Doyle and J. D. Carroll, *Routing TCP/IP*, vol. 2, Cisco Press, 2001, pp. 55–148.

[5] M. Faloutsos, P. Faloutsos, and C. Faloutsos, "On power-law relationships of the Internet topology," in *Proc. SIGCOMM*, Cambridge, MA, Sept. 1999, pp. 251–262.

[6] N. Feamster and H. Balakrishnan, "Towards a logic for wide-area Internet routing," in *Proc*. *SIGCOMM*, Karlsruhe, Germany, Aug. 2003, pp. 88–100.

[7] A. Feldmann, H. Kong, O. Maennel, and A. Tudor, "Measuring BGP pass-through times," in *Proc. PAM*, Antibes Juan-les-Pins, France, Apr. 2004, pp. 267–277.

[8] A. Feldmann, O. Maennel, Z. M. Mao, A. Berger, and B. Maggs, "Locating Internet routing instabilities," in *Proc. SIGCOMM*, Portland, OR, Aug.–Sept. 2004, pp. 205–218.

[9] T. D. Feng, R. Ballantyne, and Lj. Trajković, "Implementation of BGP in a network simulator," in *Proc. ATS*, Arlington, VA, Apr. 2004, pp. 149–154.

[10] L. Gao and J. Rexford. "Stable Internet Routing without global coordination," *IEEE/ACM Trans. Networking*, vol. 9, no. 6, pp. 681–692, Dec. 2001.

[11] T. Griffin and B. Premore, "An experimental analysis of BGP convergence time," in *Proc. ICNP*, Riverside, CA, Nov. 2001, pp. 53–61.

[12] T. Griffin, F. Shepherd, and G. Wilfong, “Policy disputes in path-vector protocols,” in *Proc. ICNP*, Toronto, Canada, Oct. 1999, pp. 21–30.

[13] T. Griffin, F. Shepherd, and G. Wilfong, “The stable paths problem and interdomain routing,” *IEEE/ACM Trans. Networking*, vol. 10, no. 2, pp. 232–243, Apr. 2002.

[14] T. Griffin and G. Wilfong, “A safe path vector protocol,” in *Proc. INFOCOM*, Anchorage, AK, Apr. 2001, pp. 490–499.

[15] S. Hares and A. Retana, “BGP 4 implementation report,” Internet Draft, Oct. 2004. (March 2005) [Online]. Available: http://www.ietf.org/internet-drafts/draft-ietf-idr-bgp-implementation-02.txt/.

[16] C. Labovitz, A. Ahuja, A. Bose, and F. Jahanian, “Delayed Internet routing convergence,” *IEEE/ACM Trans. Networking*, vol. 9, no. 3, pp. 293–306, June 2001.

[17] C. Labovitz, A. Ahuja, R. Wattenhofer, and S. Venkatachary, “The impact of Internet policy and topology on delayed routing convergence,” in *Proc. INFOCOM*, Anchorage, AK, Apr. 2001, pp. 537–546.

[18] C. Labovitz, G. R. Malan, and F. Jahanian, “Internet routing instability,” *IEEE/ACM Trans. Networking*, vol. 6, no. 5, pp. 515–528, Oct. 1998.

[19] C. Labovitz, G. Malan, and F. Jahanian, “Origins of Internet routing instability,” in *Proc. INFOCOM*, New York, NY, March 1999, pp. 218–226.

[20] Z. Mao, R. Bush, T. Griffin, and M. Roughan, “BGP beacons,” in *Proc. IMC*, Miami Beach, FL, Oct. 2003, pp. 1–14.

[21] Z. Mao, R. Govindan, G. Varghese, and R. Katz. “Route flap damping exacerbates Internet routing convergence,” in *Proc. SIGCOMM*, Pittsburgh, PA, Aug. 2002, pp. 221–233.

[22] A. Medina, A. Lakhina, I. Matta, and J. Byers, BRITE: Boston University Representative Internet Topology Generator (March 2005) [Online]. Available: http://www.cs.bu.edu/brite/.

[23] Multi-AS topologies from BGP routing tables (March 2005) [Online]. Available: http://www.ssfnet.org/Exchange/gallery/asgraph/index.html/.

[24] ns-2 (March 2005) [Online]. Available: http://www.isi.edu/nsnam/ns/.

[25] ns Manual (March 2005) [Online].

Available: http://www.isi.edu/nsnam/ns/ns-documentation.html/.

[26] J. Nykvist and L. Carr-Motyckova, “Simulating convergence properties of BGP,” in *Proc. ICCCN*, Miami, FL, Oct. 2002, pp. 124–129.

[27] D. Obradović, “Real-time model and convergence time of BGP,” in *Proc. INFOCOM*, New York, NY, June 2002, pp. 893–901.

[28] D. Pei, M. Azuma, D. Massey, and L. Zhang. "BGP-RCN: improving BGP convergence through root cause notification," *Computer Networks Journal*, vol. 48, no. 2, pp. 175–194, June 2005.

[29] D. Pei, X. Zhao, D. Massey, and L. Zhang, "A study of BGP path vector route looping behavior," in *Proc*. *ICDCS*, Tokyo, Japan, March 2004, pp. 720–729.

[30] D. Pei, X. Zhao, L. Wang, D. Massey, A. Mankin, S. F. Wu, and L. Zhang, "Improving BGP convergence through consistency assertion," in *Proc*. *INFOCOM*, New York, NY, June 2002, pp. 902–911.

[31] B. Premore, *An Analysis of Convergence Properties of the Border Gateway Protocol Using Discrete Event Simulation*, Ph. D. Thesis, Dartmouth College, 2003.

[32] Y. Rekhter and T. Li, "A border gateway protocol 4 (BGP-4)," *IETF RFC 1771*, March 1995.

[33] J. L. Sobrinho, "Network routing with path vector protocols: theory and applications," in *Proc*. *SIGCOMM*, Karlsruhe, Germany, Aug. 2003, pp. 49–60.

[34] SSFNET (March 2005) [Online]. Available: http://www.ssfnet.org/.

[35] The University of Oregon Route Views Project (March 2005) [Online]. Available: http://www.routeviews.org/.

[36] K. Varadhan, R. Govindan, and D. Estrin, "Persistent route oscillations in inter-domain routing," *Computer Networks*, vol. 32, no. 1, Jan. 2000, pp. 1–36.

[37] C. Villamizar, R. Chandra, and R. Govindan, "BGP route flap damping," *IETF RFC 2439*, Nov. 1998.

[38] L. Wang, X. Zhao, D. Pei, R. Bush, D. Massey, A. Mankin, S. F. Wu, and L. Zhang, "Observation and analysis of BGP behavior under stress," in *Proc. IMW*, Marseille, France, Nov. 2002, pp. 183–195.

[39] B. M. Waxman, "Routing of multipoint connections**,"** *IEEE Journal on Selected Areas in Communications*, vol. 6, no. 9, pp. 1617–1622, Dec. 1988.