BGP with an Adaptive Minimal Route Advertisement Interval

Nenad Lasković and Ljiljana Trajković
Simon Fraser University
Vancouver, British Columbia, Canada
{nlaskovi, ljilja}@cs.sfu.ca

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 paper, we propose the adaptive MRAI algorithm for adaptive adjustment of MRAI values. We also introduce reusable MRAI timers that independently limit advertisements of individual destinations. The modified BGP is named BGP with adaptive MRAI (BGP-AM). BGP processing delay used in the evaluation of BGP-AM is based on reported measurements. ns-2 simulation results demonstrate that BGP-AM leads to a shorter convergence time while maintaining a number of update messages comparable to the current BGP implementation. BGP-AM convergence time depends linearly on the BGP processing delay.

1. Introduction

The Internet consists of numerous heterogeneous networks without a centralized control. These networks are clustered in groups called Autonomous Systems (ASs), where each AS is controlled by a common administrative entity. Communication between ASs requires a common protocol. Border Gateway Protocol (BGP) [1] is the de facto standard inter-domain routing protocol in today's Internet.

BGP suffers from long convergence time. The BGP convergence time is the time that elapses from the moment when a change of a route occurs until all routers accordingly adjust their routing tables [2]. 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 [3], [4].

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) or MRAI round, which defines the minimum time interval between sending two consecutive update messages 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 [5]. This value is not optimal for every network topology and using smaller values may lead to a significant decrease of the BGP convergence time [2]. It has also been reported that an optimal MRAI value depends on the network topology and traffic load [2]. One proposed solution for finding an optimal MRAI value is using adaptive MRAI timers [4]. No implementation details have been reported [4].

In this paper, we propose an *adaptive MRAI* algorithm for adjusting MRAI values for every destination in each BGP router. The current implementation of MRAI timers (such as *per-peer MRAI timers*) prolongs the BGP convergence time because they impose delay on all route advertisements regardless of their destinations. Hence, we propose *reusable MRAI timers* that independently limit advertisements of individual 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 delay due to processing of BGP messages in routers (BGP processing delay) is important for analysis and simulation of the BGP dynamic behavior. A commonly used approach for calculating the BGP processing delay (such as the *uniform* BGP processing delay [2]) assumes that the average time needed for processing BGP messages depends linearly on the number of received messages. However, recent measurements [6] indicate that assuming uniform delay leads to unrealistically high estimates of BGP processing delay. We use, instead, the *empirical* BGP processing delay based on reported measurements [6], [7].

This research was supported by the NSERC Grant 216844-03 and the Canada Foundation for Innovation New Opportunities program.

We have implemented BGP-AM in the ns-2 network simulator [8]. Simulation results indicate that BGP-AM leads to a shorter convergence time, with a comparable number of update messages as the current BGP implementation [1].

The paper is organized as follows. BGP dynamic behavior is described in Section 2. In Sections 3 and 4, we describe reusable MRAI timers and the adaptive MRAI algorithm, respectively. Simulation results for two network topologies are given in Section 5. We conclude with Section 6.

2. Dynamic behavior of BGP

BGP speakers (routers) [1] exchange a large number of update messages due to persistent changes of the Internet topology. Each BGP speaker adds new and deletes unfeasible routes to destinations. Therefore, BGP is characterized by continuous changes of routing tables. During the BGP convergence process, a BGP speaker may exchange messages with its peers over several iterations until it finds the best route (converges). The end of the BGP convergence process for a single destination is defined as the instant when all BGP speakers in a network stop generating update messages for the specific destination. The BGP convergence time for a single destination is defined as the time elapsed between the instant when the first update message containing a change of the destination reachability is sent and the instant when all corresponding update messages are received [2]. To minimize the number of update messages and adequately react to changes in the Internet topology, BGP implements rate limiting by using MRAI timers. The default duration of an MRAI round is 30 s [1] and it is controlled by MRAI timers. However, manufacturers may use other values for the duration of MRAI round. For example, Juniper's default configuration sets MRAI to 0 s [6], [9].

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 [10] and the implementation of such a large number of timers is not feasible. Rather than using per-destination MRAI timers, RFC 1771 [1] proposes implementing *per-peer* timers. Each 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 corresponding to one BGP speaker. A disadvantage is that they limit advertisements of all destinations sent to the peer (even those that are advertised for the first time).

2.1 Uniform BGP processing delay

BGP processing delay is the delay imposed on an update message in a BGP speaker. It is an important parameter in the analysis of the BGP dynamic behavior. It includes the queuing time of a message and the time needed for BGP to process the received message. The uniform BGP processing delay [2] has been widely used in studies of the BGP convergence time [10]–[12]. It has also been implemented in SSFNET [13]. It assumes that a BGP speaker processes each update message independently. (Updates are queued and processed sequentially.) 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 [2]. The average processing delay $t_{BGPprocess}(p)$ for a group of N queued updates is:

$$t_{BGPprocess}(p) = N \times \frac{(p_{\max} - p_{\min})}{2}. \qquad (1)$$

Eq. (1) indicates that the average processing delay depends linearly on the number of update messages. Measurements have shown that this number may exceed 100 messages per second in the core Internet routers [10]. This suggests that using the uniform delay with p_{min} = 0.01 s and p_{max} = 1 s may not be appropriate in the case of extensive exchange of update messages. Even for a moderate number of messages (~20), using the uniform delay leads to unrealistically high values for the average BGP processing delay (~10 s).

Recent measurements on Cisco routers indicated that the average BGP processing delay is much shorter than predicted by the uniform delay [6]. They revealed that BGP speakers process groups of update messages in constant 200 ms processing cycles. When a BGP speaker operates below its maximum CPU utilization, it may process most messages that it receives at the end of a 200 ms cycle. The average BGP processing delay is between 101 and 110 ms. Over 95% 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 reach several seconds. Measurements of router CPU utilization [7] have indicated that in over 99% cases, the core Internet routers operate under 50% of their capacity. Processing BGP messages usually has a higher scheduling priority than other processes in a BGP speaker [6]. Hence, most of the time, routers process all update messages within one 200 ms processing cycle, which is significantly shorter than the value estimated by the uniform delay (~10 s).

2.2 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 [7] and the average BGP processing delay [6]. 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 [6]. 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 in all BGP speakers start at the same time, as shown in Fig. 1. Shown are times when a BGP speaker receives (dashed lines) and sends (solid lines) update messages for 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 receive new update messages regarding that particular destination.

Using the empirical delay value more accurately reflects the behavior of BGP speakers and reveals that they are often 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.

If MRAI rounds do not start at the same time, 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.

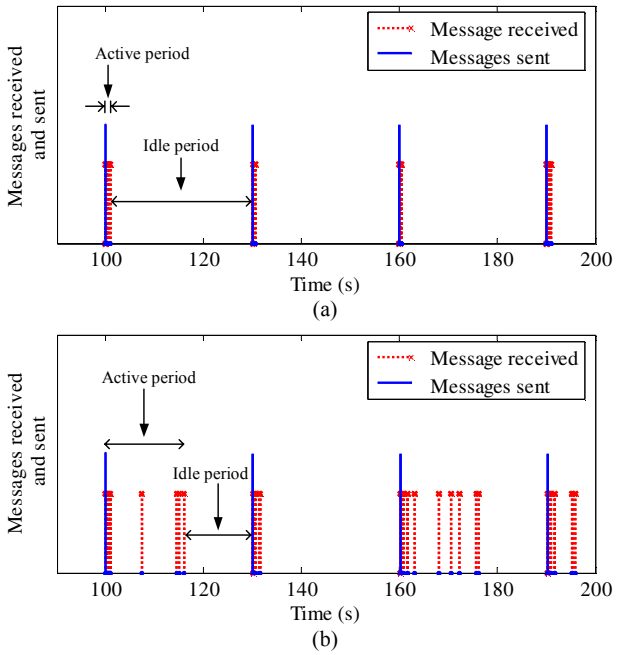

Fig. 1. Durations of the active and idle periods when using (a) empirical (200 ms cycles) and (b) uniform (pmin = 0.01 s and pmax = 1 s) BGP processing delays.

3. Reusable MRAI timers

The main drawback of per-destination MRAI timers is that they use separate timers for each destination, even for destinations that are advertised at the same time. Instead of associating one 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 redefine the rate limiting so that MRAI rounds belong to this interval (rather than being equal to 30 s). The duration of the interval defines the granularity of the MRAI round and determines the number of reusable MRAI timers. For example, if MRAI is between 29 s and 30 s, than the granularity of MRAI is 1 s and a BGP speaker needs 30 reusable timers shifted by 1 s, as shown in Fig. 2. 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 ($t_0 + 30$ s).

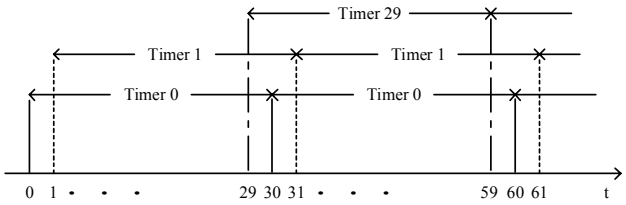

Fig. 2. 30 reusable MRAI timers with the granularity of MRAI round equal to 1 s.

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. Reusable MRAI timers enable BGP to handle advertisements independently, as in the case of per-destination MRAI timers.

The implementation of reusable MRAI timers requires storing pointers that associate each non-converged route with the corresponding timer. Similar to per-peer MRAI timers, reusable MRAI timers maintain only a list of pointers for non-converged routes. A core Internet router usually deals only with several hundred non–convergent routes, while to the entire routing table may contain over 100,000 routes [10]. Hence, the overhead for storing pointers to reusable timers is not significant.

4. Adaptive MRAI algorithm

Finding the optimal MRAI requires knowing the active period during an MRAI round. We were unable to use statistical methods to predict the active period because the distributions of the durations of active period and inter-arrival times of update messages were not available. Hence, we estimate the active period in the following 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 when an active period increases, the next round encompasses all sent update messages. We choose the margin equal 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:

$$adaptiveMRAI_{n+1}(D) = avg\_active_n(D) + 3 \times deviation_n(D), \quad (2)$$

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 Fig. 3. The variables are given in Table 1. 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.

Initial values of variables are set in the first adaptive MRAI round. In the first round, the duration and the standard deviation are set to the default value of 30 s and 1 s (rather than 0 s), respectively. Using the standard deviation equal to zero would leave the second adaptive MRAI round without the safety margin if the active period increases.

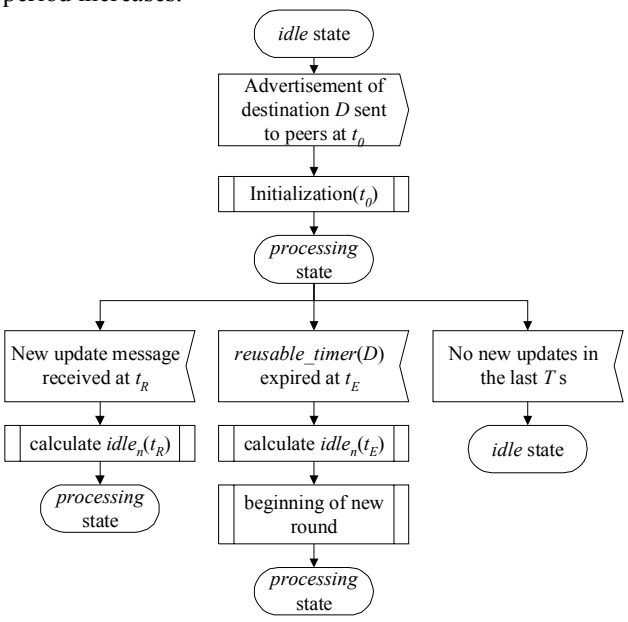

Fig. 3. The adaptive MRAI algorithm.

Table 1. Variables of the adaptive MRAI algorithm used for destination D in the n-th 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 period
$active_n(D)$	duration of the active period
$avg\_active(D)$	the average active period
$deviation_n(D)$	standard deviation of the average active period
$reusable\_timer_n(D)$	serial number of the reusable timer
$last\_received\_update_n(D)$	time when the last update message was received

Three types of events may occur in the *processing* state: *i*) reception of a new update message, *ii*) expiration of the reusable timer associated with a destination, and *iii*) the completion of the BGP convergence process. When a BGP speaker receives a new update message, the algorithm remains in the *processing* state and calculates the idle period. The idle period is 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, as shown in Fig. 4.

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 been

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 the maximum of 30 s. The maximum duration of adaptive MRAI round (30 s) is equal to the default MRAI value [1]. Hence, the adaptive MRAI algorithm cannot lead to longer BGP convergence time than the current BGP implementation. 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.

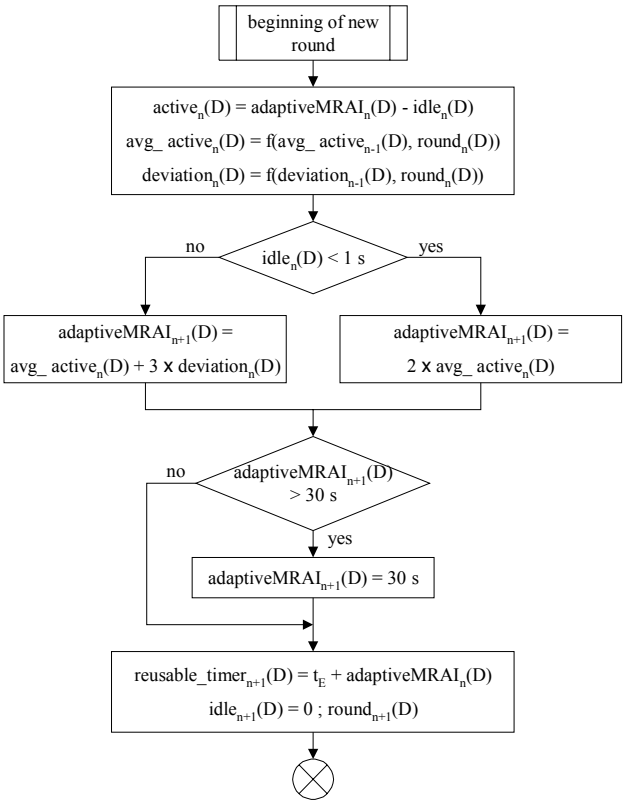

Fig. 4. The procedure for calculating the variables at the beginning of an adaptive MRAI round.

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 pre-defined time period *T*.

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.

The implementation of the *adaptive MRAI* requires the BGP speaker to store four variables for each non-converged route: $round_n(D)$, $avg\_active_n(D)$, $deviation_n(D)$, and $last\_received\_update_n(D)$. $round_n(D)$ is an integer counter, while the remaining three variables contain time stamps in milliseconds. Hence, it is sufficient to use four integers for storage. Other variables listed in Table 1 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).

The time complexity of the adaptive MRAI algorithm is $O(n)$. It is defined as the number of operations performed when the algorithm is in the *processing* state, as shown in Fig. 3. The idle period and the variables in Table 1 are calculated when a reusable timer expires (the beginning of an adaptive MRAI round). Only the idle period is calculated when a new update message is received.

The granularity of MRAI rounds determines the shortest duration of an MRAI round. The upper bound on the number of adaptive MRAI rounds for each route that has not converged is equal to the granularity. Thus, the number of adaptive MRAI rounds depends linearly on n. The time complexity of the calculation at the beginning of a new adaptive MRAI round can also be expressed as a function of the input size n. For example, if the granularity is 1 s, there is at most one adaptive MRAI round per second for each non-converged route.

During one MRAI round, a BGP speaker sends up to two update messages (one advertisement and one withdrawal) regarding each route. A BGP speaker cannot send more than one advertisement due to the rate limiting. It also cannot withdraw the same route twice in a single 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. Since the number of peers is constant for one BGP speaker, the time complexity of the idle period calculation is $O(n)$.

The average active period ($avg\_active_n(D)$) in round n is:

$$avg\_active_n(D) = \frac{\sum_{i}^{n} active_i(D)}{n} = avg\_active_{n-1}(D) + \frac{\Delta_n}{n}, \quad (3)$$

where $\Delta_n = active_n(D) - avg\_active_{n-1}(D)$. Hence, the average active period may be calculated using the current value of active period ($active_n(D)$) and the value from the previous round ($avg\_active_{n-1}(D)$). The standard deviation of the *active period* ($deviation_n(D)$) may also be calculated using the value from the previous round ($deviation_{n-1}(D)$) and Δ_n:

$$deviation_n(D) = \sqrt{deviation^2_{n-1}(D) + \frac{(\Delta_n^2 - deviation^2_{n-1}(D))}{n}}. \quad (4)$$

Based on (3) and (4), 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 calculating the variables at the beginning of an adaptive MRAI round is $O(n)$.

5. Performance of the adaptive MRAI

We implemented the adaptive MRAI algorithm, reusable MRAI timers, and the empirical and uniform delays in the ns-2 network simulator (ns-2.27) [8] using the ns-BGP 2.0 [14] module. We use two network topologies to evaluate the performance of BGP-AM and compare it with the current implementation of BGP. The simulations were performed on a 2.8 GHz Intel Pentium 4 processor with 2 GB of RAM using Linux Red Hat 9.0 operating system. The average durations of one simulation for the network topology with 110 nodes were ~2.5 min and ~2 min, for BGP and BGP-AM, respectively. The simulated topologies were limited to 110 nodes, due to extensive simulation times.

Several simplifications were adopted in order to observe the effects of the adaptive MRAI algorithm on the BGP convergence time: *i*) long-term instabilities (route flaps) were not considered because they may cause long route suppressions and, hence, mask effects of the adaptive MRAI algorithm [15], *ii*) routing policies were not applied because they may cause persistent route oscillations [16], and *iii*) each AS consisted of a single BGP speaker in order to limit the number of BGP speakers in simulations.

Commonly used scenarios for analysis [2], [10] and measurements [3], [4] of the BGP convergence time consist of the *up* (advertisement) and *down* (withdrawal) phases. 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 and the instant when all BGP speakers have found 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).

The adaptive MRAI algorithm limits the number of update messages for each destination independently. Hence, we use only one destination connected to the origin in each simulation. The impact of other update messages on the average BGP processing delay is modeled by the empirical delay. Each simulation scenario is repeated 30 times using 30 unique random number generator seeds 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 work continuously, which mimics the observed behavior of BGP speakers [4]. BGP speakers use the sender side loop detection (SSLD) mechanism and limit only the advertisements (withdrawal limiting is not used) [1]. We also assume that the BGP convergence process is completed if no update messages are exchanged between BGP speakers within the chosen period T (60 s). If not otherwise stated, the adaptive MRAI employs 30 reusable MRAI timers (with granularity of 1 s) and a 200 ms processing cycle for the empirical delay.

5.1 Completely connected graph

The simulated network is a completely connected graph with 15 nodes. This topology, although not a realistic representation of the Internet, is considered as the worst case scenario for the BGP convergence time in the down phase because it has the maximum number of possible paths for a given number of nodes [3]. ASs situated in the core of the Internet are heavily connected and their interconnections may be modeled as completely connected graphs [2], [10]. 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.

The average convergence time for BGP with the default value of MRAI (30 s) is 181.3 s, as shown in Fig. 5(a). It decreases to 45.1 s with BGP-AM. The average number of update messages is 1,480 and 1,553 for BGP and BGP-AM, respectively, as shown in Fig. 5(b). The results for BGP are similar to previously reported [2], [10]. In the case of the adaptive MRAI algorithm, the BGP convergence time and the number of update messages are independent of the MRAI round duration specified in the current BGP implementation (BGP-AM adaptively adjusts durations of MRAI rounds, whereas the current BGP uses a constant MRAI round through the entire BGP convergence process).

BGP convergence time for various durations of the BGP processing cycle is shown in Fig. 6(a). The minimum value of the processing cycle of 200 ms corresponds to the normal operation of a BGP speaker. The maximum 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. In the case of the adaptive MRAI

(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 BGP convergence time increases proportionally with the increase of the average processing delay. The average number of update messages is similar in both cases (~1,500), as shown in Fig. 6(b). It is consistent with the values shown in Fig. 5(b). The duration of the BGP processing cycle does not significantly affect the number of update messages.

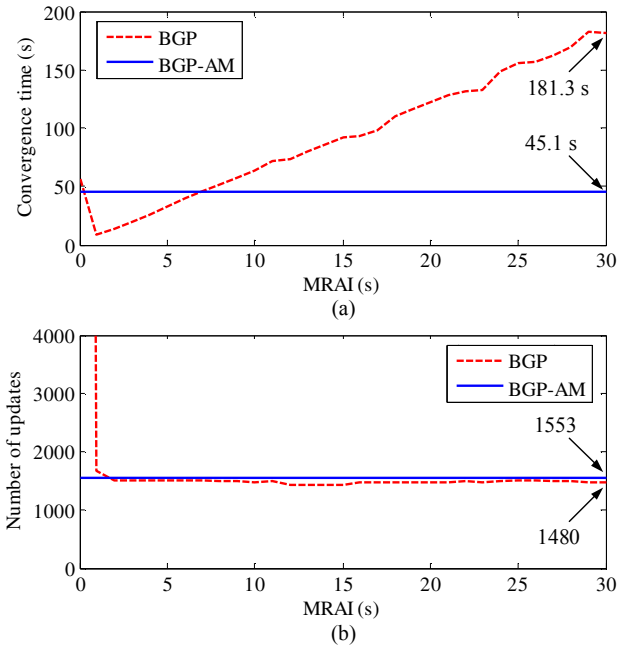

Fig. 5. Effect of MRAI on the down phase: (a) convergence time and (b) the number of update messages.

The average BGP convergence time and the average number of reusable MRAI timers for the down phase are shown in Table 2. The number of reusable MRAI timers determines the granularity of MRAI and the minimum 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 is a trade-off between complexity and performance. Table 2 indicates comparable BGP convergence times for granularities finer than or equal to 1 s. Using granularities coarser than 1 s prolongs the BGP convergence time because the maximum active time is at most 1 s. Thus, for this simulation scenario, the optimal granularity is 1 s. It achieves similar performance with fewer timers than for granularities of 0.5 s or 0.25 s. Using granularity of 1 s requires only 30 MRAI timers, comparing to several hundred per-peer MRAI timers needed in the core Internet routers [10].

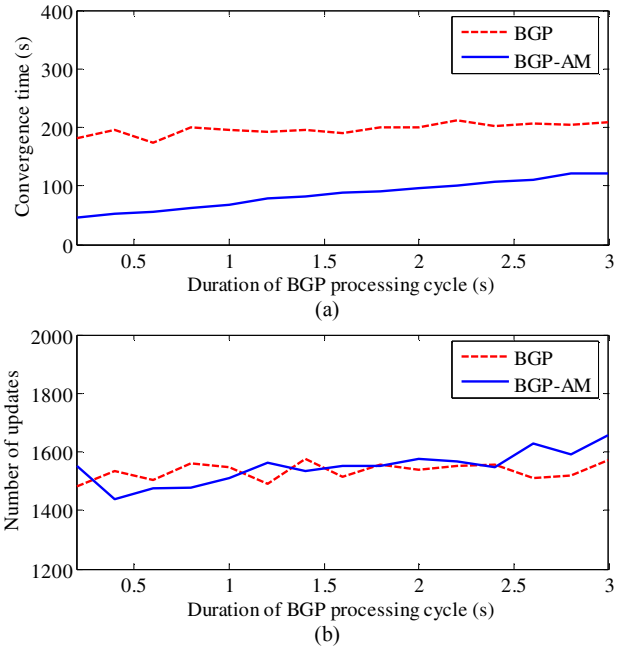

Fig. 6. Effect of the BGP processing cycle on the down phase: (a) convergence time and (b) the number of update messages.

Table 2. The average BGP convergence time and the average number of update messages for various numbers of reusable MRAI timers.

Number of timers	Granularity (s)	Convergence time (s)	Number of updates
10		56.2	1,651.7
15	2.00	54.9	1,659.8
30	1.00	45.1	1,552.9
60	0.50	45.8	1,489.0
120	0.25	45.7	1,520.0

5.2 Network with 110 nodes

We also simulated a network with 110 nodes from the University of Oregon Route Views Project [17]. Simulation results for up phase and down phase are shown in Fig. 7 and Fig. 8, respectively. Simulations are repeated for every node as the origin because the BGP convergence time depends on the length of paths from the origin to other BGP speakers.

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 algorithm decreases the BGP convergence times from ~120 s to ~35 s in the up phase, as shown in Fig. 7(a). It increases the overall number of update messages by ~30%, as shown in Fig. 7(b). In the down phase, the adaptive MRAI decreases the BGP convergence time from ~600 s to ~100 s, as shown in Fig. 8(a). The number of update messages decreases by ~20%, as shown in Fig. 8(b).

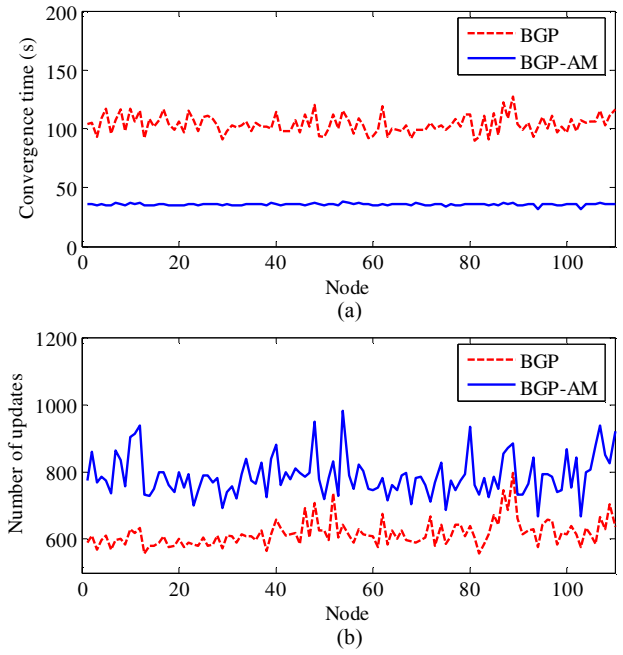

Fig. 7. Up phase: (a) the BGP convergence time and (b) the number of update messages for various origin nodes.

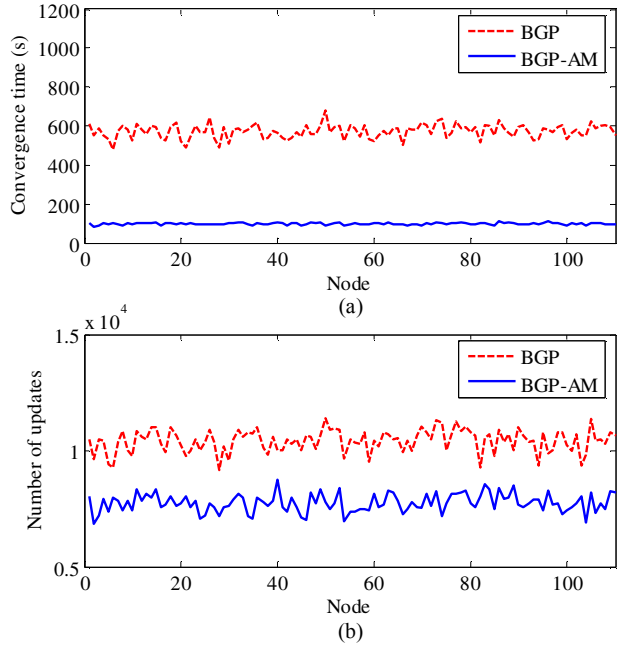

Fig. 8. Down phase: (a) the BGP convergence time and (b) the number of update messages for various origin nodes.

6. Conclusions

In this paper, 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. The proposed BGP-AM algorithm was implemented in the ns-2 network simulator. The simulation results show that adaptive MRAI leads to shorter BGP convergence times in both up and down phases for two simulated network topologies. The number of exchanged update messages remains comparable to the current BGP implementation. The BGP convergence time with adaptive MRAI is a linear function of the average BGP processing delay. As in the case of BGP, it depends on the number of MRAI rounds required for BGP to converge. The improvement of BGP-AM over the current BGP is particularly noticeable in the case of large networks in the down phase, when the BGP convergence time decreases by ~80%.

References

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

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

[3] 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.

[4] 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.

[5] S. Hares and A. Retana, "BGP 4 implementation report," Internet Draft, Oct. 2004. [Online]. Available: http://www.ietf.org/internet-drafts/draft-ietf-idr-bgp-implementation-02.txt.

[6] 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.

[7] 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.

[8] ns-2 [Online]. Available: http://www.isi.edu/nsnam/ns.

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

[10] B. Premore, "An analysis of convergence properties of the Border Gateway Protocol using discrete event simulation," Ph. D. Dissertation, Dartmouth College, 2003.

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

[12] 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.

[13] SSFNET [Online]. Available: http://www.ssfnet.org.

[14] 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.

[15] 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.

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

[17] The University of Oregon Route Views Project [Online]. Available: http://www.routeviews.org.