Dept. of Information Engineering and Computer Science

Master's Degree in
Computer Science

Final Dissertation

BGP, Catch the noise

A study on the noise detectors of BGP and their correlation

Supervisors
Renato Antonio Lo Cigno
Timothy G Griffin

graduating student
Milani Mattia

Accademic Year 2019/2020

In collaboration with the University of Cambridge

Ringraziamenti

...thanks to...

Contents

Summary

...summary....

Minimum Route Advertisement Interval (MRAI)

1 Introduction

FiXme: Expand the concepts

What we call Internet is a single network that interconnects more than 60 000 Autonomous System (AS) that shares their knowledge in order to spread the subnets reachability. An AS is a single entity that holds and controls some *IP* prefixes used by one or more operators. Every AS is responsible for its own interconnectivity policies that controls the information obtained by the outside and the one from the inside.

The propagation of the knowledge between different AS is controlled by Border Gateway Protocol (BGP). Is possible to configure the policies inside BGP in order to apply different choices to respect the constraints imposed by contract between the ASes.

BGP is growth a lot from its first release, it now include multiple optional parameters that are actually more mandatory than optional. It is the only protocol actively used on the Internet for the spreading of *IPv4* and *IPv6* networks.

Multiple of its parameters play a central role in it, and the correct setting of them could influence the performances of, not only the single AS but the entire network. For this reason the research is still active to find new technologies and trade-off between, for example, convergence time and messages transmitted.

But there are almost no studies on the correlation of those parameters. Even if the effects of some of them trigger other parameters.

1.1 Internet nowadays

- Use today studies to show how internet is today

FiXme: Maybe use some apnic data

1.2 Correlation between variables and convergence

BGP is an intrisically noisy protocol, those noises could be due to different factors. Could be BGP itself to produce noisy situations There are some parameters of BGP that point to intercept and limit noisy situations. The first one is MRAI, which is a timer with the goal to compress multiple messages, in order that a single AS can produce less messages.

Forget about the possibility to converge in seconds or even sub-seconds when we talk about internet routing convergence there are a lot of factors that influence it. The convergence time is mostly affected by some timers that rules the Internet. It could require up to different minutes to achieve a complete convergence, spread a new routing information to all the nodes.

One of the most effective timers is MRAI and it has been already proven FiXme: Insert citation that whith

1.3 Goal of this thesis

Is important to study the correlation between those parameters because Internet is not built on just one of them. Tuning one parameter is important to understand the effects on the others, and viceversa.

This thesis has the goal to gives a useful help to the understanding of this protocol interoperation on a more global scale.

2 BGP state of the art

BGP is the protocol used to control the information spreading on the Internet. It is at the version 4 published in 2006 with the Request For Comment (RFC) 4271 [1]. BGP is a Path vector (PV) protocol, it distinguish itself from the Distance Vector (DV) and Link State (LS) protocols with the major difference that it shares other than the knowledge of a path also the path itself to reach the destination.

BGP has two major sub-cateogies with a difference in the flow of information direction:

- ***Exterior BGP (eBGP)***, we talk about eBGP when the flow of information goes from the inside of the AS to other ASes;
- ***Interior BGP (iBGP)***, we talk about iBGP when the flow of information goes from the outside of the AS to the inside of it, to make aware of the new route also the internal routing protocol.

My main interest in this thesis is for the eBGP part of the protocol, for now on I will refer to it talink in general of BGP without further distinguish from the internal protocol.

When there is an interconnection between two ASes that creates a BGP link, I will talk about peering refering to thi connection, and those BGP speakers interconnected will be the peers. Each BGP connection is based on a direct Transmission Control Protocol (TCP) connection. On this links every AS can configure its own policies that would be used to evaluate routes at the reception or at the output FiXme: Find a better word. There are three different possible type of relations that can be created by two BGP speaker, accordingly to the Center for Applied Internet Data Analysis (CAIDA):

- ***customer-to-provider (c2p)*** This relationship higlights the fact that lower AS pays a hiher level AS to get connectivity and access to the Ineternet;
- ***peer-to-peer (p2p)*** This relationship is used to share the knowledge between two ASes of their customer providers without paying a higher level AS;
- ***sibling-to-sibling (s2s)*** This relationship defines the connection between ASes under the same Internet Service Providers (ISP).

During this thesis I will only consider the first two relationships c2p and p2p, the schema in FiXme: Insert figure shows how the traffic is affected based on the type of the link crossed.

As showed by the flows with a different color in FiXme: insert figure a single AS will share information considering the receiving link. If something cames from one of its customers it will share the knowledge with every other link that it has (always respecting the output policies). If a route has come from its provider or a peer then it will be only shared with its own customers. Those policies are dictated by convenience, an AS has all the advantages when other ASes decide to use it to reach a specific destination.

This behaviour can be modeled with a variant of the Stratified Shortest Path (SSP) algebra described in [2]. This is the same algebra that will be used in Chapter 3 to describe link relationships.

2.1 BGP

Once a BGP node has estabilished a connection with an other peer it will start to exchange routes with that neighbour, always respecting the policies.

Every BGP nodes has a Routing Information Base (RIB) as data structure to keep the information about the received routes, the alternative routes and what should be exported. The RIB is divided into 3 sections.

- ***ADJ-RIB_in*** This RIB contains all the routes that have been received by other AS in order to be evaluated;
- ***LOC-RIB*** This RIB contains all the best routes that have been chosen from the *ADJ-RIB_in* from the node;
- ***ADJ-RIB_out*** There is an output RIB for every neighbour of the node, it contains the route that should be advertised to the specific node.

One of the most important part of BGP is its decision process, that would be applyed to discern between the routes in the *ADJ-RIB_in* in order to update the *LOC-RIB* and, if necessary also the *ADJ-RIB_out*. The decision process is composed by three parts:

1 Calculation of the preference: This function is called every time there are new rechability information that needs to be evaluated, it will assign/update the preference value at every route in the *ADJ-RIB_in* using policy filters preconfigured. If a route doesn't respect the policy filters it will be then marked as ineligeble, otherwise a *PREF_VALUE* will be calculated and assigned to the route.

2 Route selection: This function is called at the end of the first phase, it collects all the eligeble routes and evaluate them removing routes that would create loops or that creates conflicting situations. The evaluated routes are then ordered by the *PREF_VALUE* and then the best route will be then installed in the *LOC-RIB*. In case of ties there is an algorithm that can be used to break them.

3 Route dissemination: This function can be called in different situations, it will use the information in the *LOC-RIB* to populate every *ADJ-RIB_out*, according to configuration policy.

The decision process is also responsible for the route aggregation and information reduction. At the end of the third phase the BGP speaker will execute the *Update-send* process, that is responsible for the effective dissemination.

There are multiple types of packets that can be sent by a BGP speaker, but we will focus only on the advertisement (ADV) packets. The ADV packets are responsible for the dissemination of the information to other nodes that will analyze and use the attribute inside the message to assign a preference value to the route. In particular there are two section of the ADV messages that will contain additive information and subtractive information. We can distinguish ADV messages using the type of information that are transmitting:

- ***UPDATE***, this type of messages represent the distribution of new reachability information, a new route to a destination will be shared through an update message;
- ***WITHDRAW***, this type of messages are distributed when a node want to share that it doesn't know how to reach a destination a destination anymore.

Inside those packets there are different attributes that permit to transfer information about the route (advertised or withdrawd). There is an attribute that describe the address that the route represents, another one that contains the path that will be used to reach the destination, the next hop used, etc. During the year multiple new RFC have introduced, modified, updated, removed attributes that can be found inside an advertisemente message. Not all the attributes are mandatory for BGP nodes, infact for a node is possible to receive a route with attribute that it is not able to interpret but (if configured to do so) it will share the route with also the unknown attributes.

Is important to remember that all those information are useful for the policy filter that every node can have, for example some nodes would automatically discard any route that contains a specifi AS in the received path.

The *Update-send* process is responsible for the distribution of the messages that are stored in the *ADJ-RIB_out*. It should execute again some checks on the RIB, removing unfeasible routes and remving routes that has already been advertised to the pear. It also has to respect a temportal constraint,

introduced in [1], a BGP speaker can't send to the same neighbour routes for the same destination more often than the MRAI value. MRAI act as a timer which goal is to avoid continous update storm caused by decision changes in some peers in the network.

Another property that can be found in BGP nodes that affects the messages transmitted is the Implicit Withdraw (IW). This property permits to reduce the number of messages tha are distributed. Without this option when a BGP node discover a new besth path to reach a destination should send a withdraw followed by an update to its neighbourhood. Thanks to this option is sufficient to send just the update, the other nodes will learn that the best path is changed simply looking to the previous alternative and comparing them.

2.2 BGP Noise

There are basically two type of noises in BGP, the inherent noise and the noise caused by external sources. Those two type of noises are triggered by different causes but are substantially not discernible one another.

The first one is caused by the protocol itself when it tryies to convergence on new knowledge. The sharing of new routes can cause new ADV that can then trigger the *Path Exploration* problem. This is a noise caused by the protocol itself that acts as eco chamber for new best paths that changes untill the best possible path is taken in consideration.

One BGP parameter tries to limit this noise acting as a message chache with a compression system. This parameter is MRAI and it permits to avoid sending a message for every new one received using a timer. Only the best decision after that time will be shared.

The second type of noise is caused by a source outside the protocol. A missconfiguration, a faulty interface can cause the send of not necessary messages. For example the withdraw and the advertisement of a route at continous intervarl. This type of behaviour will cause continuous storms of messages and the triggering of the first noise type.

BGP introduce a parameter with the RFC 2439 [3] that is called Route Flap Damping (RFD) to overcome this behaviour. This paramter increases a value every time a flap or a route change are detected, when this value passes a predefined threshold the route will be suppressed. This value will alway decay using an exponential decay function, even if it doesn't overpass the threshold. Once the route has been suppressed the BGP speaker must wait enough time that the value goes below an other threshold before sharing it again.

Those two parameters are clearly correlated one another from the fact that one triggers the other and vice versa, is possible to create particular topology that has different performances based on the values assigned to those parameters. Think about two clique networks connected one another by only one node, this node will act as a bottleneck, probably its RFD threshold would be easily overpassed and then it depends on it decay function before it can send again the network to the second clique triggering MRAI on those nodes that will experience the path exploration problem.

FiXme: I think that the graphs in Fig 2 and Fig 3 of Fabrikant et al. in [4] will easily trigger both MRAI and RFD

2.3 BGP MRAI

MRAI is one of the parameters that mostly affect the convergence of BGP A high value of it could unnecesarly delay the transmission of messages, but in the opposite case a value too small can provoke a lot of messages, one for every decision change of the node. There are a lot of studies about it, and it has already been showed that the number of messages and the convergence of time can depend on it [5]. It has been also already proven by Fabrikant, Rexford et al. [4] that an incorrect configuration of it could lead to tremendous consequences.

MRAI has been introduced in the 4th version of BGP [1] and it is nowadays a mandatory argument for every BGP node, otherwise the load in terms of messages to process and decision changes would be incalculable. Its main purpose, as anticipated in Section 2.2 is to prevent, or at least mitigate the noise created by BGP itself.

At the base of MRAI there is a timer that controls how much time must pass between one ADV

and the following one. The timer is peer-based, for each interconnection an AS could have a different MRAI, but it acts for every destination in parallel, this means that there is a different timer for each destination that a node would share for every BGP relations that it has. The idea behind it is taht in this time the BGP node will be able to receive other possible routes, evaluate them and send a decision on the best route considering multiple alternatives. It has the property to compress the input messages sequence in order to have an output message sequence with a smaller number of ADV.

The use of MRAI is as follow every time the route selection proces change a route in the *OUT* RIB of a neighbour:

- If there isn't an active MRAI timer for the destination change send the ADV and set an MRAI timer.
- If there is an active MRAI timer for the destination then don't send anything.
- When the active MRAI timer ends if there still is the necessity to send an ADV the send it and set an other MRAI timer.

The second passage permits to the route selection process to be executed multiple times before the actual transmission of the decision. Thats because MRAI limits only the transmission and not the decsion process. The condition to the last passage is due to the fact that the compression some times could lead to the unnecessity to actually send a message.

The default value defined in the RFC 4271 for MRAI is equal to 30 s. But, MRAI is a really controversial parameter, it has received multiple revesions and studies. In 2008 there has been a proposal to reset it default value to 5 s [6] thanks to different studies that takes in consideration the dimension of the topology and the latency [7]. In 2010 a proposal RFC of *IETF* [8] says that the default would be left to the arbitrary choice of the operators and that withdraw message could compleatly avoid it. But this freedom would damage the convergence and the number of messages distributed as shoed by Fabrikant et al. [4].

Is clear that MRAI affect the network performances, but what affects MRAI and, by consequences, the performances? Obviously the choice itself of a different MRAI strategy, as showed for example in [9] where the centality was used to obtain better results in case of network faults. But, giving the fact that the main function of MRAI is to compress the input messages also the sequence of messages recept could be meaningful. Giving that MRAI is a link-based parmeters also the number of links that a node has could influence it, and by consequences the position in the topology. A well connected node will be more likely to receive multiple paths and messages than one with only one link.

2.4 BGP RFD

RFD is a parameter introduced in BGP to overcome the problems caused by the exterior sources of noise. Its main function is to avoid fluctuating routes to overload BGP nodes with continous message storms. It has been introduced with the RFC 2439 in 1998 [3]. Also RFD is a controversial parameter, it has been studied and revaluated different times, but recent studies showed that the majority of the operators still use deprecated values from 2001 [10]. Furthermore, other studies shows that the majority of the ADV that thravels through the Internet are genereated by a restricted set of AS but RFD seems to be to restrictive and affect the majority of the ASes traffic [11].

RFD will use a single value, called *figure of merit*, to evaluate the actual situation of a route, while this value evolve the RFD algorithm will take decision on what to do. The evolution of the *figure of merit* is dictated by the messages received, with fluctuations it will grow, while, over time, it will use a quadratic decay function to make it decrease. Fluctuations, or flaps, are represented by the reception of the withdraw and the announcement of a route, a path change is also considered a flap, even if thanks to IW is limited to just one ADV.

There are other parameters that are part of this BGP component, the more importants are presented in Table 2.1.

Other than the name of the parameters in Table 2.1 is showed also the default value decide by cisco. The RFC 2439 [3] gives some guidelines on how to set those values but actual choice is left to the discretion of implementors. The first three parameters, *Withdraw, re-advertisement, attribute change*

Parameter	Cisco default values
withdrawal penalty	1.0
re-advertisement penalty	0.0
attribute change penalty	1.0
suppress threshold	2.0
half-life (min)	15 (900s)
Reuse Threshold	0.75
Max Suppress Time (min.)	60 (3600s)

Table 2.1: RFD parameters

represent the penalty applied to the *figure of merit* when the omonim event happen. The *suppression threshold* represent the level at which the BGP node will suppress the route and don't advertise it untill the figure of merit goes below the *reuse threshold*. The decay of the *figure of merit* follows a quadratic decay function which rate is calculated using the *Half-life* parameter. Other than those parameter that defines the evolution of the filter the *Max Suppress Time* will override all of them, because a route cannot be suppressed for more than that time, it doesn't matter the figure of merit accumulated by this route.

An example of the figure of merit evolution could be see in Figure 2.1, This image has been taken from [10].

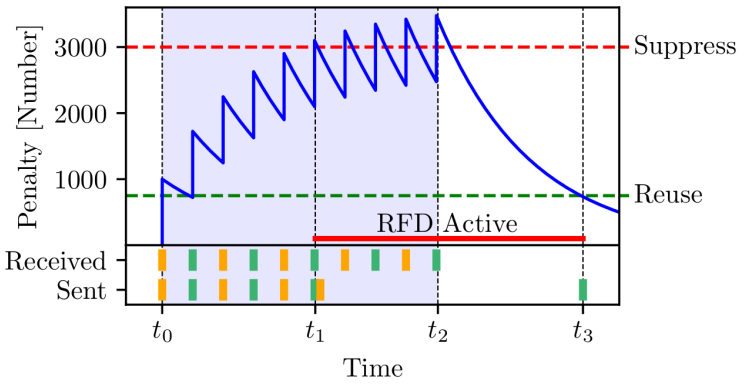


Figure 2.1: Example of evolution of the RFD *figure of merit* taken from [10], yellow messages represent withdraws and green ones are advertisement, dashed lines are the suppression and reuse threshold

Figure 2.1 shows an hipotetic evolution of the RFD filter, it doesn't relay on the default value of the cisco implementation. Is possible to see in the lower part of the plot the messages received by the BGP speaker, yellow one represent withdraws while the green are announcements. Is possible to see that the penalty value grows at each flaps and as soon it reaches the suppression threshold the route will not be advertised to any neighbour. While after the decay has reached the reuse level is possible to advertise the route again.

Is possible to see that RFD doesn't make any difference on its own on what is causing the flaps, it simply reacts to the actual situation of the network. Is not even possible to determine where is located the flap, if is the source that is flapping havily for some reasons or an AS in the middle of the path that is malfunctioning.

RFD has a troubled history, maybe even more than MRAI. In 2006, thanks to the pubblication of [12], RIPE-378 [13] recomends to disable it. Few years later the pubblication of the article from Pelsser et al. [11] RIPE and IETF shares that says that now RFD should be used with the updated parameters [14, 15]. Unfurtunatly, the study from Gray et al. [10] in 2020 shows that the majority of the AS uses RFD with outdated parameters from the RFC 2439.

FiXme: Add an end

2.5 Topologies

FiXme: Talk about the underlaing topology of BGP nodes that is not the actual geographical topology

The study of the tomografy of the internet has been a really "hot" topic in the 90's. Is not possible to know exactly the Internet network, thats because the edges of the network are defined by commercial contracts between the nodes that are kept private between the parts.

We are not compleatly blind, we know some properities of the network. Internet is a hierarchical topology where there are different layers of nodes interconnected one anothere and different types of nodes. The nodes in the highest level, Tier one nodes, are the nodes that interconnect all the lower layers of the network and them are connected one another in a clique network.

3 Discrete Event Simulator

Experiments on BGP are not applicable on the Internet, for this reason different studies show their results using a simulate environment [5] FiXme: Insert other citations. The majority of the studies uses small graphs and each node of the graph simulate the behaviour of a BGP speaker. Each node represent also a single AS and the BGP speaker is it's own exterior router, for simplicity, reduced to one speaker that handles all the connections.

For this reason, I decided to use and expand a Discrete Event Simulator (DES) that permits to have different grades of freedom, respecting on the other side all the properties required for a reliable simulator environment. I decided to use the *Simpy*[1] package to make the environment evolve. I decided for this package for the extensive documentation and because it has been already used for different studies, demonstrating its adaptability [16, 17].

I developed the DES as a highly modular environment.

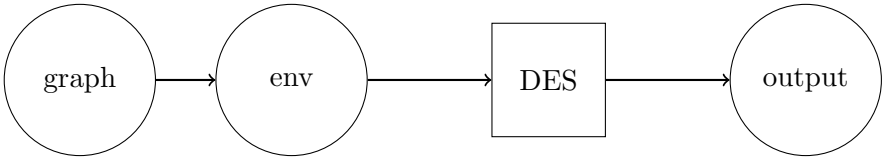


Figure 3.1: Discrete event simulator structure

In Figure 3.1 is possible to see the basic idea of the simulator. The first component needed is a graph, represented by a *graphml* file, this file is the descriptor of the network. it defines also all topological information and all the properties of every single node. FiXme: Look for a Cref implementation of this In Code 3.1 is possible to see an example of a *graphml* file, it describes that node 0 contains a single destination and that the edge between nodes 2 and 5 is controlled by the policy —2, 2, 2— that defines a servicer-provider policy. Policies are encoded using the convention described in [2].

```
<node id="0">
     <data key="d0">10.0.0.0/24</data>
</node>
<edge source="2" target="5">
    <data key="d2">2, 2, 2</data>
</edge>
```

Code 3.1: Graph example

The graph is then embedded in the environment file, this file is in *json* format and it describes how the environment is characterized, it gives the initial values for the Random Number Generator (RNG) so that each experiment is replicable and other properties, like where the output should be saved, and, most importantly how the experiment should be conducted. There is two possible evolution of the environment:

- ***Continuous evolution***: In this category all the nodes that contains at least a destination will continuously share and retrieve the destination accordingly with the distributions defined in the environment;
- ***Signaling evolution***: Is possible to define a precise signal that should be executed by the nodes that contain a destination, for example, the signal "AWA" defines that there will be an announce followed by a withdraw and another announce.

[1] Simpy website

The DES take as input this *json* file where all the information are described, it creates an object for each node in the graph file, with each own characteristics. After the initialization, all the nodes that contain a destination will schedule the first advertisement of it to their neighbour. The simulation run will terminate only if there are no more events scheduled or if the maximum simulation time is reached.

The DES will then produce a *CSV* output, with all the events that can be analyzed to see the evolution of a specific node or to evaluate the whole network.

3.1 DES Environments

Thanks to the environment codification in a *json* file is possible to define experiments with a high grade of freedom. Is possible to define multiple delays as probability functions vectors that will provide multiple runs possibility. For example, if we have 5 different possible seeds and 3 different delays, the total number of runs combinations is 15, as showed in Code 3.2. is possible to run one of the possible combinations of parameters through the identifier of the single run.

```
"simulation" : {
    // seed(s) to initialize RNG
    "seed" : [0, 1, 2, 3, 4],
    ....
    // Multiple withdraw distributions
    "withdraw_dist": [{"distribution": "unif", "min": 5, "max": 10, "int": \
   0.1},
                      {"distribution": "unif", "min": 8, "max": 10, "int": \
   0.1},
                      {"distribution": "unif", "min": 2, "max": 3, "int": \
   0.1}],
    ....
}
```

Code 3.2: Environment example

In the environment is possible to define also the processing time, this time is used inside each BGP node to emulate the processing of information or the evaluation of a packet. Though the *delay* parameter is possible to define the default delay on the edges, is important to remember that the links are FIFO so there is no reordering of messages in the same link, there is also no messages lost. That because it was out of the scope of this thesis to study the evolution of the protocol with packet loss, but it could be a future work.

3.1.1 Clique environment

One of the special environment that I used it's composed by a clique graph graph of different dimensions, an example of clique graph is given in Figure 3.2.

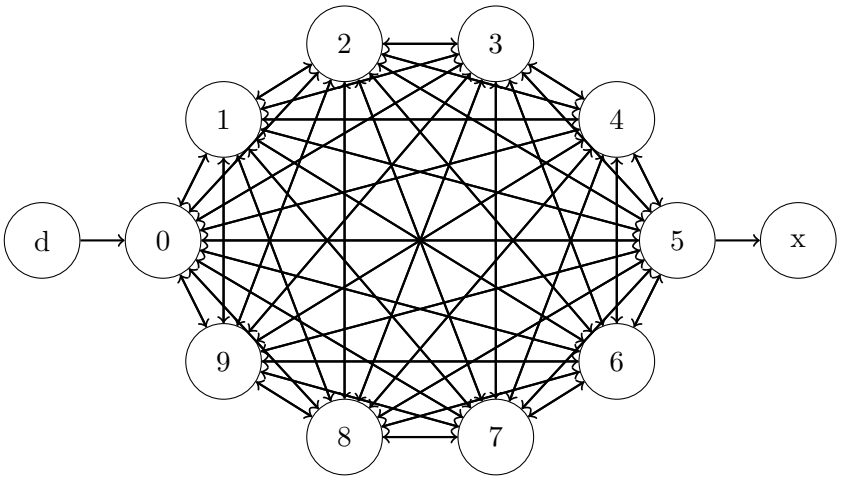


Figure 3.2: Clique graph example

The only node that shares a destination is the node "d", the node 0 will then spread the knowledge to the whole network, and the node "x" will act as a black hole for all the possible paths that the node 5 will share. This topology is used to enforce the path exploration problem.

3.1.2 Fabrikant environment

Another interesting chase to test the path exploration problem is the one presented in [4]. In that study, Fabrikant et al. presents how particular MRAI setting could make the network converge with an exponential behaviour because of the path exploration problem. I used the basic example of their study to investigate how the choice of MRAI is fundamental for the network convergence. An example of the network used is presented in Figure 3.3.

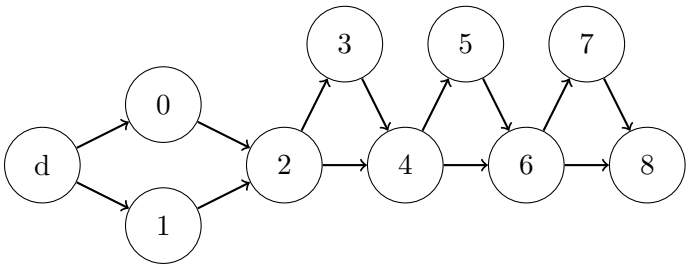
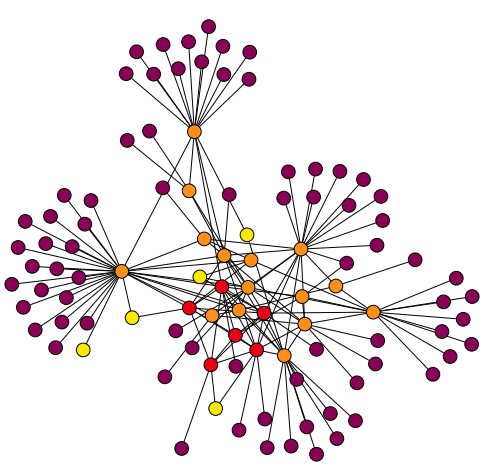


Figure 3.3: Fabrikant chain graph example

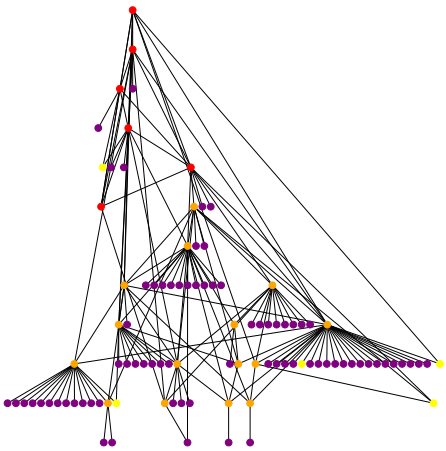
The path exploration problem is caused by the delay on the node 0-2 edge. The node 2 will receive the destination through node 1, after a small amount of time the network will converge to the best path (without using the backup links). But, after a while, node 2 will receive the network also through node 0 and it will prefer this new path, provoking than the reconfiguration of all the other nodes that will use the backup links for a while, announcing their new path. A wrong configuration of MRAI can provoke the entire exploration of the possibility set.

3.1.3 Internet-like environment

The last noteworthy environment is the one whose purpose is to simulate Internet behaviour. This has been possible thanks to the study by Elmokashfi et al. [18] and the internet like graph generator present in Networkx [2] (a python library famous for graph and network studies). An example with a small set of nodes is presented in Figure 3.4



(a) Internet like graph with an "explosive" layout



(b) Internet like graph with a "hierarchical" layout

Figure 3.4: Internet like graph colored to show the hierarchically structure, 4 types of nodes, T (tier 1 mesh), M, CP, C (Customers, purple one)

[2] Networkx internet as graph generator

The different nodes are colored accordingly with the node type represented. The tier one nodes that generate the central clique are colored in red and is possible to notice in Figure 3.4b that them are in the highest levels of the networks. This environment has been used to study the behaviour of the network with topologies resembling the real internet.

4 The Protocol as a Finite State Machine

An Finite State Machine (FSM) could be useful for a lot of purposes, to debug the protocol, to understand what is happening, to analyze leeks. It has been already done for a lot of protocols FiXme: insert citations, but not for BGP.

FiXme: Give more examples on what a protocol FSM is useful for

4.1 BGP generalization

The main idea behind the BGP FSM is to represent the knowledge as states and different set of messages as transitions. The knowledge is represented by the actual routes that the node knows on how to reach a single destination. Transitions encode the messages that a node has received to trigger the state change, on the edges are also inserted the response messages that the node will transmit. We can see an example of this transitions in Figure 4.1

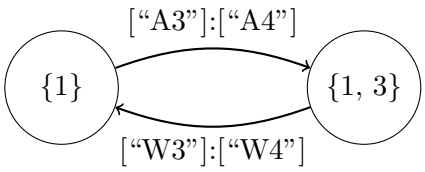


Figure 4.1: Example of the BGP FSM state transition

In Figure 4.1 there are two states, both represent the knowledge of the node, the first one represent the RIB with just the route 1, in the second state the RIB will contains both the routes 1 and 3. This transition si caused by the reception of the advertisement of the route 3 and will couse the transmission of an other advertisement. The opposite transition is caused by the reception of the withdraw of the route 4 with the consequent withdraw of the route 4.

In BGP messages transfer information about routes, there could be the advertisement or the withdraw of the route.

Thanks to MRAI the evaluation of multiple messages could be delayed and provoke then the compression of them. For this reason on the edges is possible to see multiple messages, for example "A1W1A1", that will be compressed in "A1" and then evaluated.

The concept for a BGP FSM has been expanded from [19].

4.2 BGP FSM experiments

The first experiments, about the translation of a single node evolution in a FSM, goal is to reproduce what has been shown in [19]. The graph used for the study is presented in Fig. 4.2.

This topology, Figure 4.2, present an Stable Paths Problem (SPP) with five nodes [20]. The SPP model is used to eliminate much of the complexity of BGP. The arrows in the graph represent the flow of information, node 1 is the one that will receive a new route to reach a hypothetical destination and it will spread this information through an ADV to all it neighbours. The translation to the Communicating Finite-State Machine (CFSM) will use an enumeration to encode all the paths that a single node will encounter, for example, the path "5 3 1" will be converted in *a3*, each path has its own identifier. In case of withdraw the route will be encoded as *w3*.

The properties of the environment for this experiment are listed in Table 4.1.

The total number of runs generated by this environment is 100.

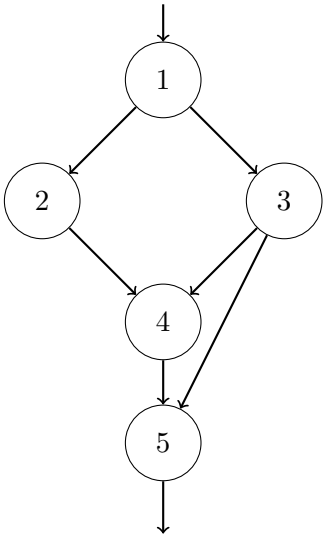


Figure 4.2: Graph from fig 4 of [19] used to study the FSM of the nodes

Property	Value
Seeds	$[1, 50]$
Signaling	"AW"
Withdraws delay	Uniform distribution between 20 s and 30 s
Announcement delay	Uniform distribution between 20 s and 30 s
MRAI	0 s for every link
Link delay	Uniform distribution between 0.001 s adn 1 s, uniform distribution between 0.012 s and 3 s

Table 4.1: FSM example environment properties

FiXme: this paragraph is cumbersome The two nodes of more interest are node 4 and node 4. The first one can receive multiple combinations of messages from node 2 and 3, for sure there will be two announcements and two withdraws because node 1 has to respect a predefined signaling. but, those messages could be reordered in different ways, and, for each sequence of them we can encounter a different sequence of output messages through node 5. Giving that the routes from node 2 and 3 will have respectively as ID 2, 3 the table Table 4.2 All possible inputs of node 4 are the shuffle of all possible outputs of nodes 2 and 3 preserving the local order.

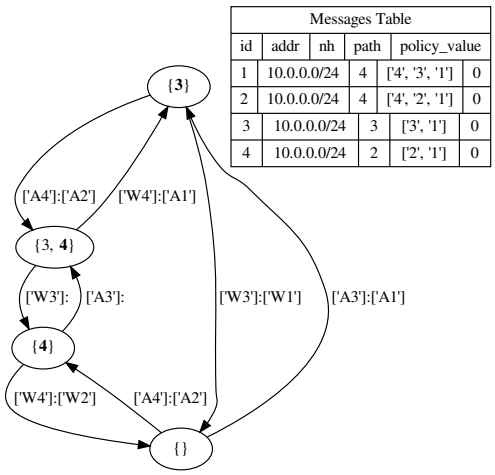
Input signal	Output signal
$a2a3w2w3$	$a4w4$
$a2a3w3w2$	$a4w4$
$a3a2w2w3$	$a5a4a5w5$
$a3a2w3w2$	$a5a4w4$
$a2w2a3w3$	$a4w4a5w5$
$a3w3a2w2$	$a5w5a4w4$

Table 4.2: Node 4 different possible inputs and output

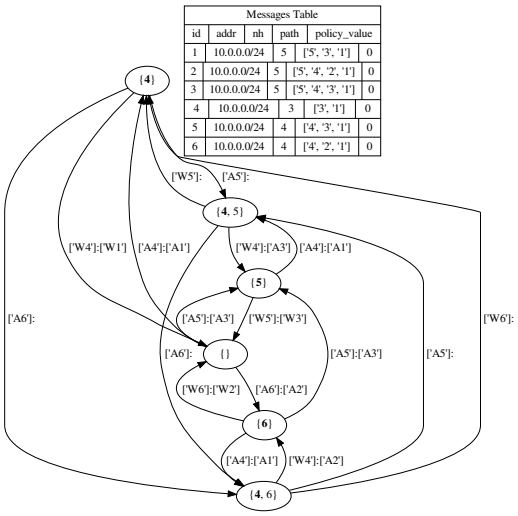
The node 5 will receive all the possible outputs from node 3 and 4 increasing the number of possible signals from 6 of node 4 up to 71 but some of them produce the same output signal, so we have in total 52 unique output signals from node 5.

From the 100 total runs we can generate the CFSM of node 4 and node 5, in order to be able to study how the nodes reacts to different input signals. The two CFSM are presented in Figure 4.3.

FiXme: Remove message table from Figure 4.3?



(a) Node 4 CFSM from the environment of Table 4.1



(b) Node 5 CFSM from the environment of Table 4.1

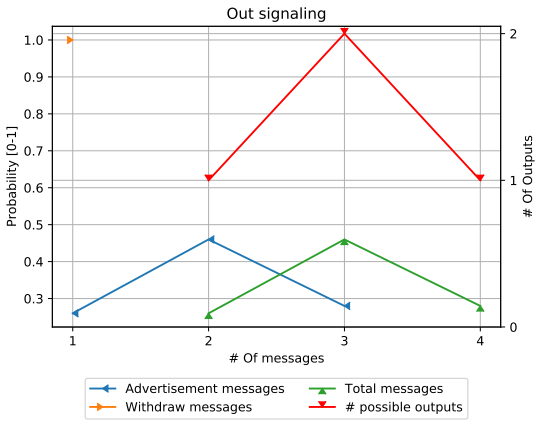
Figure 4.3: CFSM of nodes 4 and 5 of the graph Figure 4.2 with an input signal of "AW"

The states of the CFSM in Figure 4.3 are represented by the knowledge of the nodes, composed by the routes that are in the RIB of the node. The bold value is the actual best route to the destination chosen by the node. If in the state transition to a new state the best path is not affected then the node will not transmit the new route to its neighbours, for an example take a look to Figure 4.3a from the state $\{1\}$ to the state $\{1, 3\}$ where the node 4 will learn a new route that is not the best one.

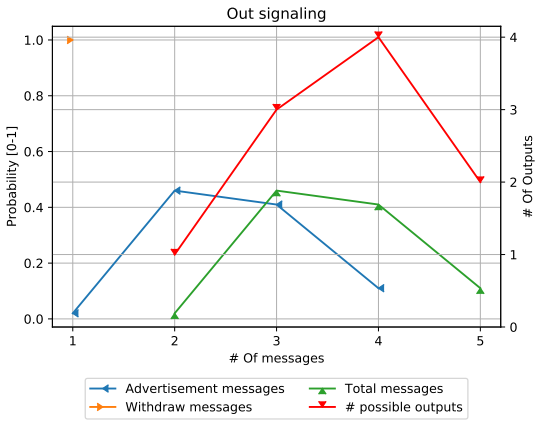
The effects of the implicit withdraw can be seen in Figure 4.3b the transition from $\{1, 4\}$ to $\{1, 3\}$ thanks to the reception of the announcement $a3$ from the node 4.

As written in [19], I would like to underline the fact that, given the 52 unique possible outputs of the node 5 it would be very difficult to infer the initial signal that provokes all the transitions.

We can also analyze those output signals, having all the events for each single run we can infer which were the most common output signals that a single node experienced. Is sufficient to take all the transmitted messages of a node and look the sequence of advertisement and withdraws.



(a) Node 4 output signals study



(b) Node 5 output signals study

Figure 4.4: Output signal study of nodes 4 and 5 of the graph Figure 4.2 with an input signal of "AW" at node 1

The plots in Figure 4.4 represents the probability of an output signal of a certain length to appear and the number of unique output signals of a unique length has been found. The x axis represents the number of messages in the output signal, a message is a single announcement or withdraw. The first y

axis represents the probability to see a certain number of messages taking a random output signal from the output. For this axis there are three lines that refer to it, the blue one represent the number of advertisement messages in the output signal correlated with the respective probability. For example, in Figure 4.4a there is a probability around 0.45 to have exactly two advertisement messages per output signal. And respectively a probability slightly larger than 0.25 to have only one advertisement or three. We can also notice that we didn't see more than three advertisements or less than one. The green line instead represents the total number of messages in the signal, without distinguishing between advertisement and withdraws. By the fact that we will always have one withdraw (the orange line) this line is simply shifted by one unit in respect of the advertisement line. The second y axis refers to the number of unique output signals encountered and their length. For example, in Figure 4.4b we will have 1 unique output signal of length 2, 3 signals of length 3 and 4 of length 4 and 2 of length 5.

Those plots do not give a complete prospective of all the possible outputs that can be generated but only the ones encountered during the runs. In fact, during the 100 runs we encountered only the output signals listed in Table 4.3.

Signal	Frequency
$a1a2a1w1$	28
$a2a1w1$	23
$a2w2$	26
$a1a2w2$	23

(a) Node 4 output signals encountered

Signal	Frequency
$a1a2a3w3$	15
$a1a3w3$	16
$a2a1a2w2$	19
$a1a2w2$	28
$a1w1$	2
$a2a1a3w3$	6
$a2a1a2a3w3$	8
$a3a1a2a3w3$	3
$a2a1w1$	2
$a3a1a3w3$	1

(b) Node 5 output signals encountered

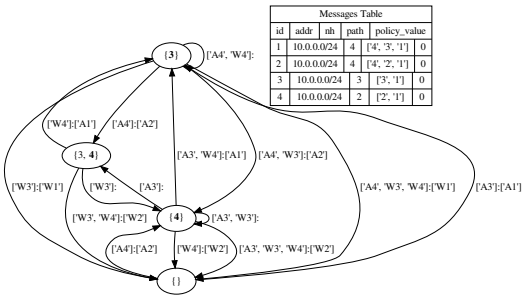
Table 4.3: Node 4 and 5 different output signals encountered during the runs

4.2.1 MRAI and BGP FSM

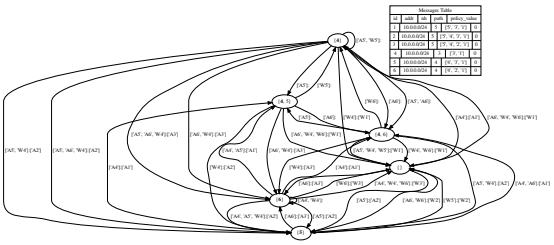
How would MRAI affect the study of the signals produced by Figure 4.2? The answer is that the number of states will be the same but the number of possible transitions will explode because there will be a lot more possible input signals that will be compressed and evaluated by the nodes.

We can see the effects of MRAI on the CFSMs in Figure 4.5.

FiXme: Figure 4.5b is not readable at all, move the two figure one after the other



(a) Node 4 CFSM from the environment of Table 4.1 with MRAI=30 s



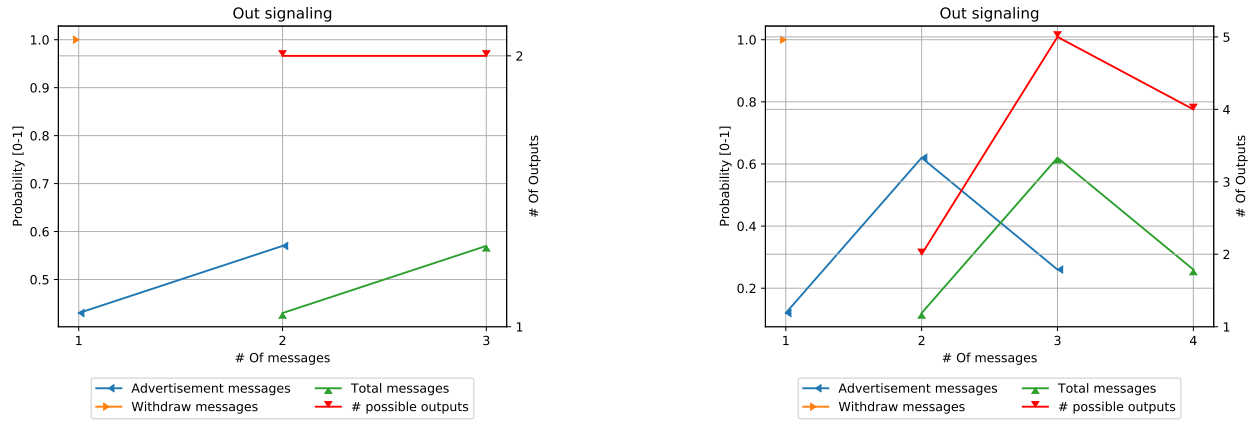
(b) Node 5 CFSM from the environment of Table 4.1 with MRAI=30 s

Figure 4.5: CFSM of nodes 4 and 5 of the graph Figure 4.2 with an input signal of "AW" with MRAI=30 s

Figure 4.3a and Figure 4.5a permits us to compare the two CFSMs of node 4 and is possible to

nice a big difference in terms of edges between one figure and the other, the first one has 8 transitions, the second one 15. For the node 5 we pass from 16 transitions to 36.

But the positive effects of MRAI can be found in the output signals, showed in Figure 4.6.



(a) Node 4 output signals study with MRAI=30 s

(b) Node 5 output signals study with MRAI=30 s

Figure 4.6: Output signal study of nodes 4 and 5 of the graph Figure 4.2 with an input signal of "AW" at node 1 with MRAI=30 s for every link

Comparing Figures 4.4b and 4.6b is possible to notice that there is a different distribution of output signals. The x axis never reach the value of 5, this means that the output signals of the node 5 never used more than 4 messages. And we can also notice that the majority of the signals this time have a length of 3 messages, instead of the previous 4. This is a hint that MRAI can have positive effects on the number of output messages produced by single nodes, having, however, more possible transitions to consider.

4.3 BGP FSM explosion

We know that MRAI is not an easy parameter, the incorrect setting of it can lead to an explosion of messages and an exponential convergence time. This problem has been studied by Fabrikant et al. [4] and the origin of the problem has been attributed to the *path exploration* problem. This is a well-known problem in the BGP community and it is experienced by a node when it enters in a transitory phase where it accepts and publishes not optimal paths towards the destination before reaching a stable state. *Path exploration* can lead to an enormous amount of messages even with a small set of nodes [21].

As we saw in Section 4.2.1 that MRAI can influence the CFSMs of the nodes and their output signals, which impact could it have if it is not set correctly?

I have then created an environment that resembles the study conducted by [4] using a topology like the one described in Section 3.1.2 with 3 rings. with different MRAI settings. The environment properties are presented in Table 4.4.

Property	Value
Seeds	$[1, 30]$
Signaling	"A", "AW", "AWA", "AWAW"
Withdraws delay	Uniform distribution between 5 s and 10 s, Uniform distribution between 10 s and 15 s
Announcement delay	Uniform distribution between 5 s and 10 s, Uniform distribution between 10 s and 15 s
Link delay	Uniform distribution between 0.5 s adn 3 s, uniform distribution between 2 s and 4 s

Table 4.4: Fabrikant experiments environment

In total, for each signaling experiment this environment produces 240 runs. I have then introduced 4 different MRAI strategies for each different signal. The different MRAI strategies are the following one:

- ***Fixed* 30 s**: MRAI is fixed for each link to 30 s;
- ***No MRAI***: MRAI is fixed for each link to 0.0 s;
- ***Ascendant***: MRAI will be doubled at each leach ($1 - 2 - 4 - 8 - ...$);
- ***Descendent***: Reverse of the ascendant case, MRAI will be divided by two at each leach.

Another important factor to consider during those experiments is the IW capability of BGP. This parameter will influence the number of messages that will be transmitted.

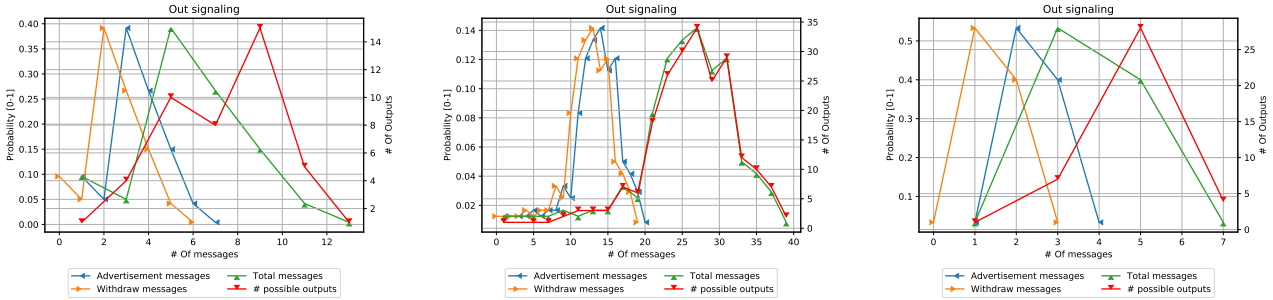
The results of all those different experiments, in terms of CFSM are exposed in Table 4.5

Signaling	IW	No MRAI		Fixed 30s		Ascendent		Descendent	
		$\|S\|$	$\|T\|$	$\|S\|$	$\|T\|$	$\|S\|$	$\|T\|$	$\|S\|$	$\|T\|$
"A"	Yes	12	19	15	26	7	12	16	24
	No	30	100	30	125	9	21	30	132
"AW"	Yes	52	181	37	103	24	71	40	80
	No	51	221	57	263	22	90	58	274
"AWA"	Yes	51	170	25	50	33	148	50	137
	No	69	364	37	180	30	203	66	419
"AWAW"	Yes	77	461	38	132	54	300	53	148
	No	78	500	62	429	48	350	66	441

Table 4.5: Fabrikant CFSMs results, $|S|$ is the dimension of the states set $|T|$ is the dimension of the transitions set, The worst results for each category are colored in gray, the topology contains 3 rings, as Figure 3.3

As is possible to see from the grey squares in Table 4.5 the more complex CFSMs are the ones without MRAI and with a descendent MRAI timing. The second case is the same described in [4] and the extremely high number of transitions is caused by the *Path Exploration* problem. Is also noticeable that the IW has a huge effect on both the number of states and the number of transitions. This because there are less possible combinations of input signals for the nodes. The opposite case in respect of the *Descendent* strategy obtains great results, even better than the actual standard of 30 s for each link. This performance improvement is caused by the fact that each leach will wait enough time to have more information from its predecessor in order to have more information to make the best decision.

The *Path Exploration* problem is also noticeable evaluating the output signals of the last node of the chain. Results about the output signal of the node 8 (the last node of the gadget) are presented in Figure 4.7.



(a) Node 8 output signals study with ***Fixed* 30 s** strategy

(b) Node 9 output signals study with ***Descendent*** strategy

(c) Node 9 output signals study with ***Ascendant*** strategy

Figure 4.7: Output signal study of nodes 8 of the graph Figure 3.3 with an input signal of "AWA" at node d with the ***Fixed* 30 s**, ***Descendent*** and ***Ascendant*** strategies, without the help of the IW

The first signal study, Figure 4.7a is the one that represents the actual standard of the protocol [1]. We can notice in that particular output study that the maximum detected length of a signal is 13 and it's the last probable output, while the most probable output length is 5. While we can notice the *Path Exploration* problem by the spike of unique output signals with a length of 9, this mean that the node experienced some changes in its decisions. The worst-case scenario is the one represented by Fig. 4.7b where the maximum length of the output signal reaches almost 40 messages, but the most probable output signal has a length between 20 and 30. This is the marker of a lot of decision changes in the best path for the node 8. Opposite to that case, we found the *Ascendent* strategy in Figure 4.7c where the number of output signals never used more than 7 messages. The node 8 in this last case almost never experienced the *Path Exploration* problem, thanks to the fact that most of the times the information it receives from the neighbourhood are already corrected.

In conclusion of this chapter, we can say without doubts that MRAI influences the number of states experienced by a node and, confirming what has been sad in [4], that an incorrect setting of it can lead to an explosion on the number of states and transitions. It is also noticeable that a different setting of MRAI can also lead to a better scenario than the standard one. Alternatives to the standard MRAI has been already presented FiXme: Include citations and maybe find a better end of the chapter

5 BGP MRAI dependency

MRAI is one of the parameters that mostly has caused divergences in the scientific community. And, after the introduction in the protocol since version 4 [1] FiXme: Check this sentence, is one of the more studied for the possibility to improve the protocol or generate exponential convergence behaviour in small network [4].

The protocol strictly depends on this parameter, because as we saw in Chapter 4, the incorrect use of it can lead to tremendous consequences, even worst of not having it at all. In other cases, with a particular setting of it is possible to improve the network performances. Recent studies about centrality metrics on routing protocols introduce, through the distributed computation of the metric, to a timer trade-off improvement [22, 23]. This kind of approach has been also applied on BGP with positive results on network failures [9, 24].

All those study points out how we can set MRAI to improve network performances, but what about how MRAI reacts on different problems? Is it possible that MRAI reacts differently based on where the signal occurs? In fact, our hypothesis is that is not enough just look to the MRAI setting because also other factors can be relevant. For example, a change near the central clique of T nodes could provoke a large storm of messages because MRAI doesn't affect in time the spreading of information. While, a change in the periphery could be cushioned without it reaching the center of the network.

5.1 Clique graph

The clique topology is one of the worst-case scenarios as specified in Labovitz et al. [25] I used two approaches in this Environment, the first one keeps the IW active the second one doesn't use of this property. To emphasize the effects of this parameter with the effects also of different MRAI settings.

The Environment properties are listed in Table 5.1

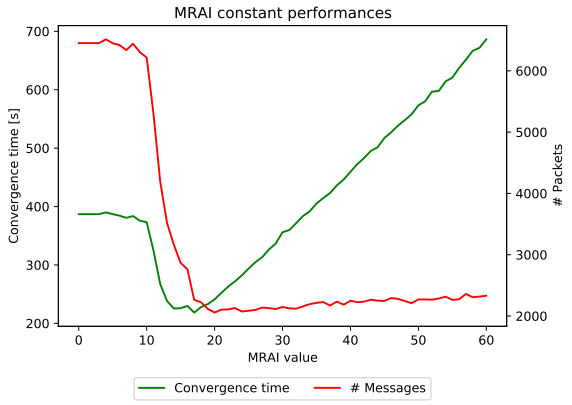
Property	Value
Seeds	$[1, 10]$
Signaling	"AW"
Withdraws delay	Uniform distribution between 1 s and 5 s
Announcement delay	constant distribution of 5 s
MRAI	$[0, 60]$
Link delay	Uniform distribution between 0.0001 s and 0.5 s

Table 5.1: Clique environment properties

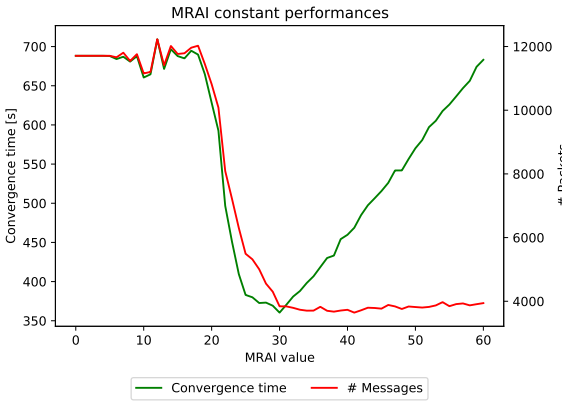
As described in Table 5.1, for each MRAI value has been executed 10 different runs of the environment. The clique graph used in this experiments is composed by 15 nodes. The MRAI strategy used is the *fixed*, so every link will have the same MRAI value. The results are presented in Figure 5.1

Is possible to notice in Figure 5.1 both the effect of MRAI and IW. Those plots represent the network performances in terms of convergence time and number of messages transmitted to reach the convergence after the transmission of the signal "AW". The convergence time is represented by the average time from all the nodes in the network. Each point in the plots is the average of the 10 runs with the *fixed* MRAI value on the x axis. The left y axis should be used with the convergence time, the green line, while the second y axis represents the number of messages transmitted, the red line.

The effects of the first one are present in both the plots but in two different moments. In Figure 5.1a MRAI affects both the convergence time and the number of messages around 20 s up to 30 s. After the threshold of 30 s, the effects of MRAI are counterproductive, the convergence time is negatively



(a) Network performances **with IW**



(b) Network performances **without IW**

Figure 5.1: Evolution of the network performances on the clique graph of 15 nodes using a fixed MRAI from 0 to 60 seconds. FiXme: use the same interval in the y-axis?

affected because the nodes start to wait more time without obtaining more useful information. This can be seen also in the number of messages that reaches a constant value.

In Figure 5.1b we can see the same effect but with a higher MRAI value. The number of transmitted messages reaches the constant value with an MRAI value around 30 s. The effects of IW can be saw also in the number of messages and the convergence time with a low MRAI, is possible to reach even 12 000 messages while with IW the maximum value is around 6500 messages.

5.2 Internet like graph

The internet like environment is more complex than the clique one, but it permits to have a more close vision of what can really happen on the Internet. During my studies, I used different topologies with 1000 nodes resembling the Elmokashfi properties [18] already described in Section 3.1.3.

Using this graph I will look for a possible correlation between MRAI and other factors that can influence the network. First of all MRAI has a dependence on how it is set, I'm going to compare different MRAI strategies that can be used on an Internet-like graph. Another influencing factor could be the signal used as an input or even the position of the node that provoke the change.

5.3 Strategy dependence

Like I mentioned before, the network performances depend on the MRAI strategies chosen. For this reason, the first goal of my study is to point out these differences. In order to do that, the first study that I would like to present is the one that studies how the standard protocol evolves on an Internet environment.

The property of the environment chosen are described in Table 5.2

Property	Value
Seeds	$[1, 10]$
Signaling	"AW"
Withdraws delay	Uniform distribution between 1 s and 60 s
Implicit withdraw	Active
MRAI	$[0, 60]$
Link delay	Uniform distribution between 0.012 s and 3 s

Table 5.2: Internet like environment properties

The graph is an *Internet-like* graph with 1000 nodes. The node that will execute the signal has been chosen randomly between all the nodes of type "C". This graph will be the same for all the

experiments in this section.

For each MRAI strategy, that I'm going to present, has been executed 61 experiments, one for each possible value of MRAI, for each experiments thanks to the environment variable has been executed 10 runs. In total for each MRAI strategy has been run 610 different runs

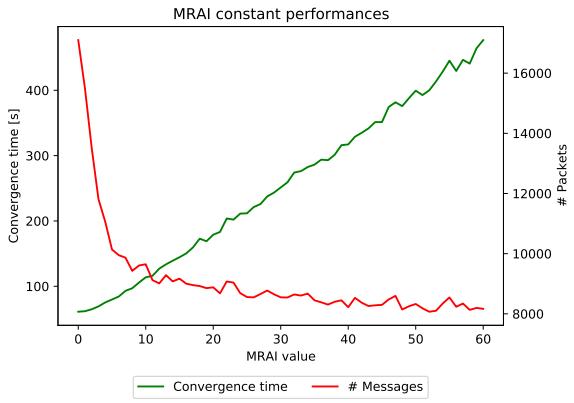
As MRAI strategies I decided to use the following two:

- ***Fixed***; Every link will have the same MRAI value;
- ***DPC***; This strategy assign a different MRAI value to each link depending on the centrality of the node [24]

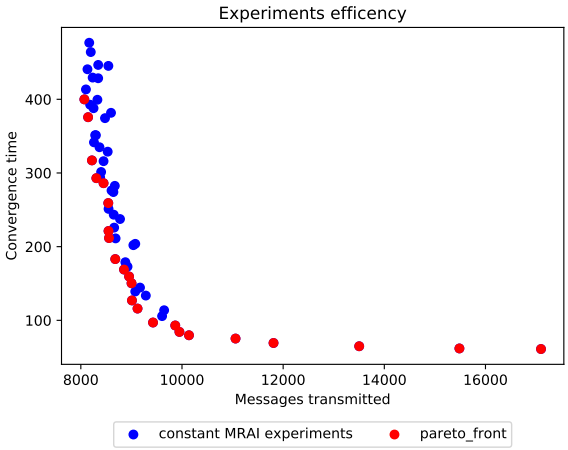
The centrality metric used is called Destination Partial Centrality (DPC) and thanks to the fact that has been already demonstrated that is possible to calculate it in a distributed way [9] I will assume that it is calculated in advance and that every node knows it's own centrality to set the timers.

To permit a comparison between those two different strategies a constraint on the MRAI assignment has been introduced, the *mean* of all the timers in the network must be equal between the two strategies. For the *Fixed* strategy, this is constraint intrinsically respected. For the *DPC* strategy, the timers are multiplied by a factor k that permits to keep the average equal.

The results of the first strategy are showed in Figure 5.2.



(a) Network performances, messages VS convergence time with different MRAI values



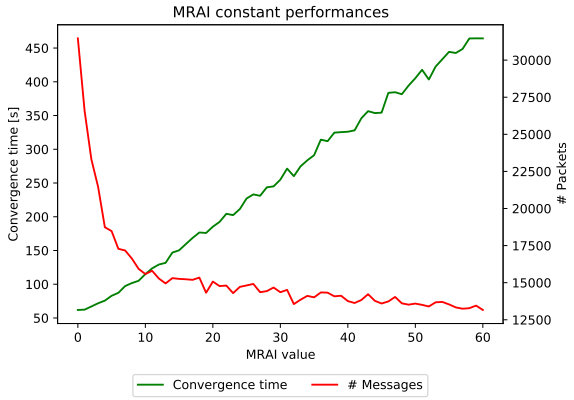
(b) Pareto front of Messages VS Convergence time

Figure 5.2: Evolution of the network performances on the **Internet Like** graph of 1000 nodes using a fixed MRAI from 0 to 60 seconds.

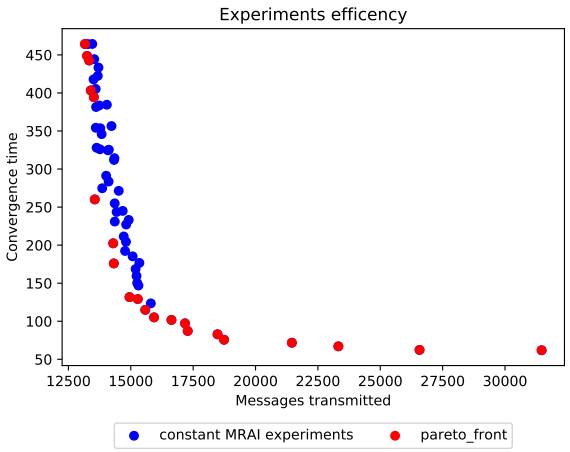
As is possible to see in Figure 5.2a without MRAI we would have a low convergence time, dictated mostly by network delays and processing time. With, on the other hand, an enormous amount of messages. Slightly increasing the MRAI value, the number of messages will fell down reaching a constant value around 8000, while the convergence time continuously grows linearly, as it happened for the clique graph in Figure 5.1. This continuous linear grow is dictated by the fact that nodes keep meaningful information for more time before sharing them with their neighbourhood. Figure 5.2b represent the Pareto front of those experiments. The Pareto frontier is the set of values that are Pareto efficient, this concept has been already used in engineering to define the set of best outcomes from the trade-off of two different parameters [26]. We can clearly see that the majority of the points is concentrated on the left of the chart, this means that few MRAI values would give as a result a high number of messages and a small convergence time. While multiple MRAI values would concentrate around the same value of messages transmitted. This can confirm the fact that MRAI would not influence messages after a certain threshold but only the convergence time.

The results of the same environment without IW are showed in Figure 5.3.

Also in this case, comparing Figures 5.2 and 5.3, is possible to notice that IW helps to reduce the number of messages and the convergence time without impacting the network performances trend.



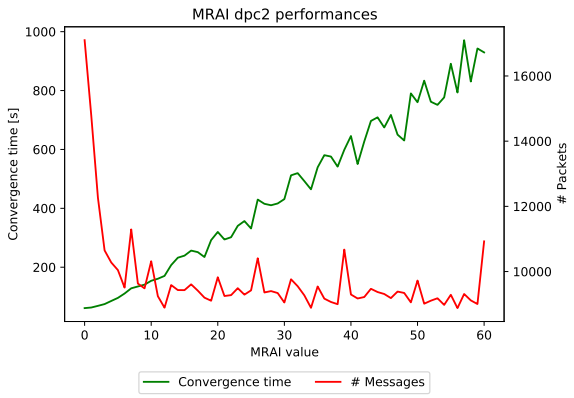
(a) Network performances, messages VS convergence time with different MRAI values



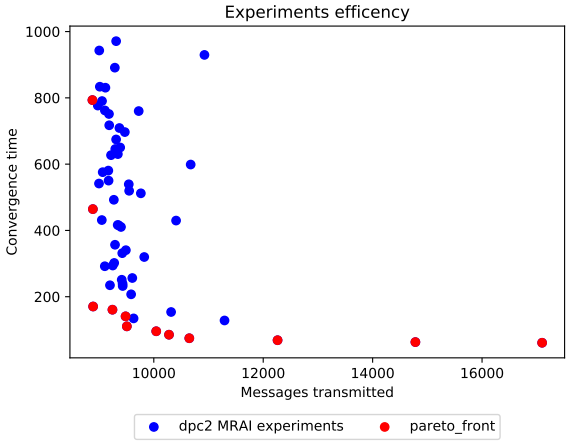
(b) Pareto front of Messages VS Convergence time

Figure 5.3: Evolution of the network performances on the **Internet Like** graph of 1000 nodes using a fixed MRAI from 0 to 60 seconds. **Without IW**

The second strategy, the one dependant on the DPC, produced the results in Figure 5.4 As mentioned before, all the timers are adjusted to respect the same mean as in the *fixed* MRAI experiments. For this reason points with the same MRAI value are comparable to one another.



(a) Network performances, messages VS convergence time with different MRAI values



(b) Pareto front of Messages VS Convergence time

Figure 5.4: Evolution of the network performances on the **Internet Like** graph of 1000 nodes using a *DPC* MRAI strategy with an $MRAI_{mean}$ from 0 to 60 seconds.

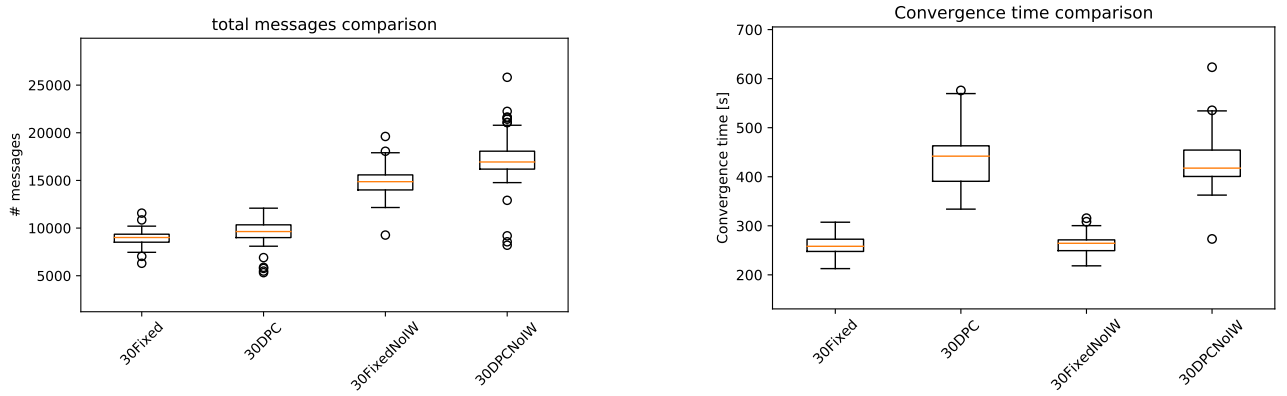
FiXme: Cumbersome, reading again it's not very clear what I'm explaining This second strategy leads to the performances showed in Figure 5.4, is possible to notice that the number of messages transmitted fell down very quickly and it reaches the convergence value around an MRAI value of 10. But, it is also noticeable that there are a lot more spikes in this trend, that deviate more from the constant value around 9000 messages. Also, the convergence time is affected by this behaviour.

FiXme: Consider introducing a figure to show both trend in the same plot Comparing Figures 5.2 and 5.4.

Is possible to notice that the two strategies lead to a different trend. Both are equal at the beginning with MRAI equal 0 but, after a while, both the number of transmitted messages and the convergence time diverge. The number of messages with the DPC strategy variate more and it converges around 9000 messages, while the *fixed* strategy reaches 8000 messages. And the convergence time with the second strategy grows more quickly. This is caused by the central clique of tier-one nodes that have a high MRAI value. The high MRAI value is caused by the fact that all the leafs has 0.0 as centrality that cause an MRAI value of 0 and to respect the $MRAI_{mean}$ value the central nodes needs a huge MRAI. For example, with an $MRAI_{mean}$ of 30 s the node 1 (that is one of the central clique nodes)

has an MRAI value of 79.35 for all its neighbours.

The standard value of MRAI is 30 s as described in [1] so I compared those strategies performances in a box-plot in Figure 5.5. I decided to run 100 different runs for each strategy with the $MRAI_{mean}$ fixed to 30 s.



(a) Network performances, messages necessary to reach convergence with different MRAI strategies

(b) Network performances, time required to reach convergence with different MRAI strategies

Figure 5.5: Network performances comparison with different MRAI strategies, Graph internet like with 1000 nodes, MRAI value 30 s, number of runs for each strategy 100

In Figure 5.5 we can compare those two strategies, the first figure, Figure 5.5a represent the number of messages transmitted by the 100 runs, we can see that the two strategies, without IW, are really close to one another. While in the time required for convergence, Figure 5.5b there are some huge difference between the two strategies, is not negligible that with the DPC strategy the time required is almost the double of the standard time.

In conclusion, we can say that the MRAI strategy is one of the factors that can influence the Network performances.

FiXme: Maybe I can introduce more strategies to expand this section

5.4 Pareto Efficiency Front

The strategies exposed in Section 5.3 are just few of the possibilities that are available. For this reason, I would like to explore the set of possibilities looking for MRAI configuration randomly generated.

I would then study the space of possibilities that are generated through the Pareto efficiency plot and compare the results with the Pareto efficiency graphs. To permit this comparison I would set MRAI randomly but like for the DPC strategy respecting the average required.

The environemnt used for those experiments is showed in Table 5.3

Property	Value
Seeds	$[1, 10]$
Signaling	"AW"
Withdraws delay	Uniform distribution between 1 s and 60 s
Implicit withdraw	Active
MRAI mean	$[0, 60]$
MRAI values	Uniform distribution between 1 s and 120 s
Experiments per MRAI mean	10
Link delay	Uniform distribution between 0.012 s and 3 s

Table 5.3: Random MRAI environment properties

Thanks to this environment I'm going to run in total more than 600 compleate experiments. For each MRAI mean value I will generate 10 different graph with a random assignment of MRAI for each

link. I wil then execute 10 different runs for each radom graph that will produce the average result of 1 experiment. The total number of experiments is 610.

in Figure 5.6 is possible to see all the 601 points generated.

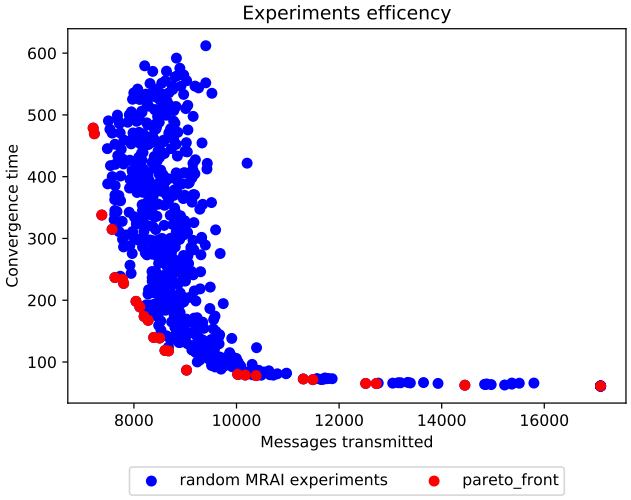


Figure 5.6: Pareto front generated by 601 experiments on an internet like topology with 1000 nodes, MRAI generted randomly and adapted to the mean

As we can see the trend in Figure 5.6 is similar to the one that we saw for the same signal in Section 5.3. For the majority of configuration the number of messages transmitted is never over 10 000 but the time required to converge grows continuously.

In Figure 5.7 is present a comparison between the random experiments, the fixed MRAI strategy and the DPC strategy from Figures 5.2b and 5.4b

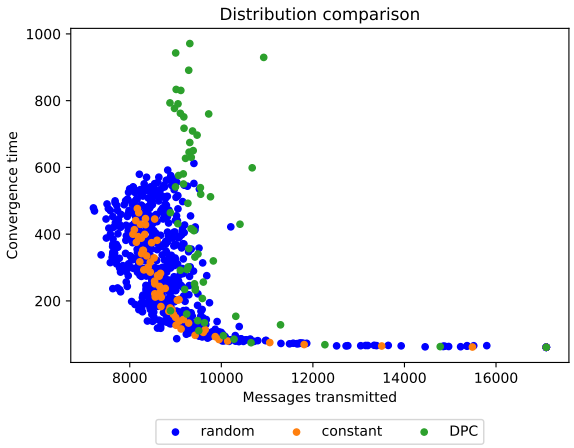


Figure 5.7: Pareto front generated by 601 experiments on an internet like topology with 1000 nodes, MRAI generted randomly and adapted to the mean, vs fixed MRAI strategy and DPC MRAI strategy.

As we can see in Figure 5.7 all the strategies have the same behaviour, but is also possible to see that the random strategy is the only one with experiments that produces less than 8000 messages. This is important because it a prove that there are better possibilities rather that the classical one.

For this reason MRAI can be tuned to have a better trade-off between number of messages transmitted and convergence time.

5.5 Signal dependence

I would like to analyze how much the signal can impact the convergence performances with the two different strategies of Section 5.3.

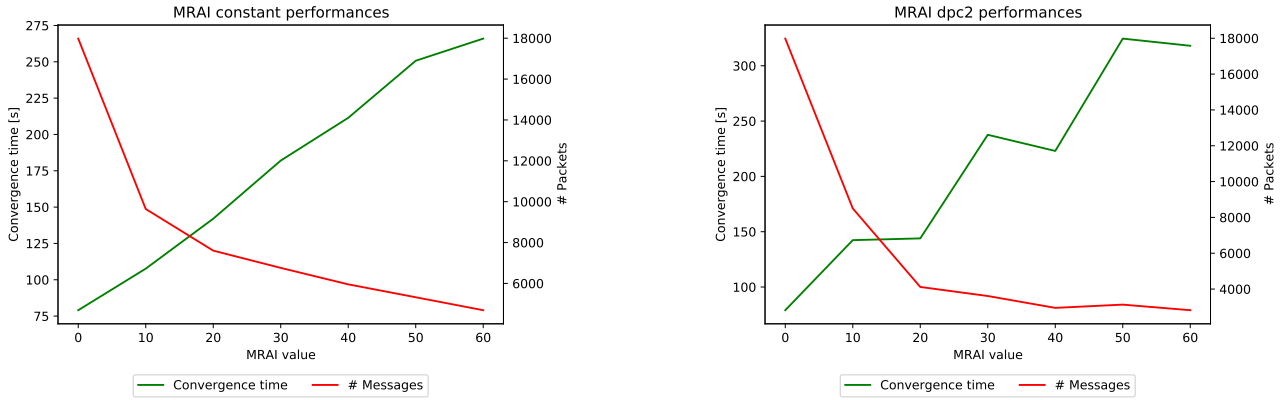
For this reason, I used the same environment described before and execute the experiments with different input signals from the same node, “AWA”, “AWAW” and “AWAWA”.

In those experiments plays a role also the “*re-advertisement distribution*” for the second and third “A”, it has been set to a uniform distribution between 1 s and 60 s, like the “*withdraw distribution*”.

For those experiments, I didn’t evaluate the case with IW deactivated. FiXme: Explain why

FiXme: all the plots has an MRAI steep of 10 s to give a hint on the trend, redo the plots with a step of 1 s

In Figure 5.8 is possible to see the evolution for the signal “AWA”.



(a) Network performances, *fixed* MRAI strategy

(b) Network performances, DPC MRAI strategy

Figure 5.8: Network performances comparison with different MRAI strategies, Graph internet like with 1000 nodes, signal “AWA”

Is possible to notice in Figure 5.8 a huge difference in respect of the plots in Figures 5.2 and 5.4. The DPC strategy was able to outcome the standard *fixed* strategy over multiple prospective. Analyzing Figure 5.8b is possible to notice that the red curve, the one that refers to the number of messages transmitted has a very fast fell, with an average MRAI timer *of* 30 s the number of messages is less than 1/4 in respect of an MRAI *mean* of 0 s. The convergence time curve has a completely different trend in respect of the previous experiments. We can notice some steps trend. This is caused by the fact that now the timer is able to effectively act on the signal. MRAI doesn’t affect the first message, in this case the first “A” of the signal, but it can affect the next two messages. In fact, some nodes are able to cache both the “WA” part of the signal and completely avoid sending anything at all, because they have already transmitted the first “A”. The complete compression of the signal “AWA” is “A”. The other evolution, for the “AWAW” and “AWAWA” signals, are showed in Figures A.1 and A.2

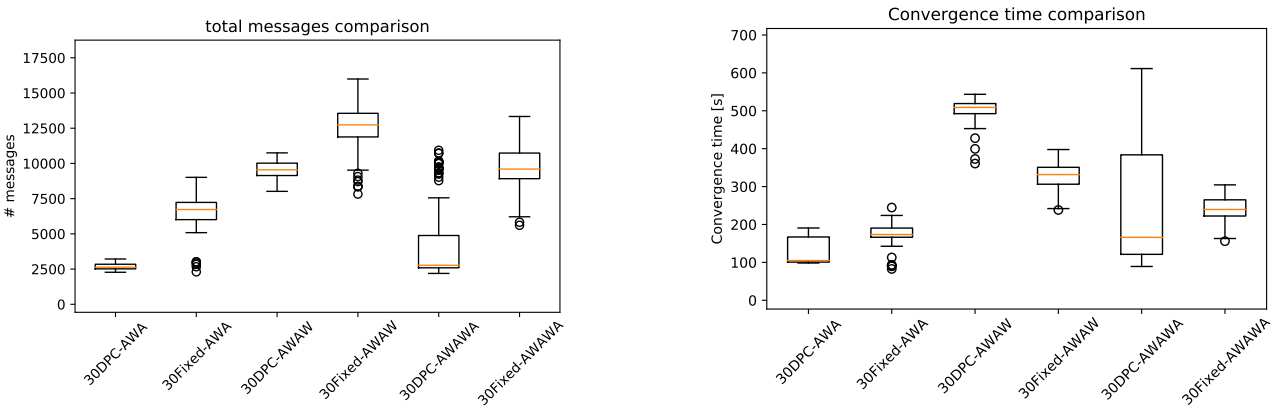
Like before, comparing the standard 30 s fixed MRAI I executed 100 different runs for each strategy and each different signal, the results are exposed in Figure 5.9b.

Is possible to notice in Figure 5.9 that both the strategies have different performances in respect of the signal produced by the single node source. In particular, performances are better when the signal ends up with an “A”. That’s because, after the first “A”, giving the MRAI timer long enough, a node is able to compress a sequence that ends with another “A” to the empty set and don’t send anything more. While if the sequence ends up with an “W” it has to, at least, send another message to notify the withdraw.

Other than that is possible to notice that the DPC techniques has better results in terms of messages transmitted, while it could have a higher convergence time. This is caused like before by the high MRAI values used by the most central nodes.

In conclusion, there is a correlation between MRAI and the sequence of messages transmitted by the source node. In particular more the timer is able to compress sequence more the performances are good.

FiXme: consider moving figures from the appendix to the chapter



(a) Network performances, messages necessary to reach convergence with different MRAI strategies

(b) Network performances, time required to reach convergence with different MRAI strategies

Figure 5.9: Network performances comparison with different MRAI strategies, Graph internet like with 1000 nodes, MRAI value 30 s, number of runs for each strategy 100, signal "allSignals"

5.6 Position dependence

The last factor of influence for MRAI that I would like to study is how much the position of the signal source can influence the convergence. The main hypothesis is that a node closer to the central clique, that generates a signal would provoke a message storm bigger in respect of a node on the perimeter of the network. This is true only if MRAI is large enough to block the storm near the source of it exporting only the correct information at the end of it.

5.6.1 Different signal sources

As first try I have decided to try 10 different destination chosen randomly on the same graph, this graph is an Internet like topology with 1000 nodes. After that, I run the same environment with all the different destination. I also used different MRAI strategies, repeating the experiments for all of them. With this results is possible to analyze how different signal sources provoke different network performances and also study how different MRAI strategies adapt to different nodes that provoke messages storms.

The

5.6.2 Hierarchical influence

What about the position in the hierarchy? Internet is very strong hierarchical graph, Figure 3.4b is an example with a small set of nodes but it is possible to define different levels of the graph. If we take the central clique as the root of the graph then all the nodes will be at a certain distance (in terms of hops) from it.

Nodes that are on the same hierarchical level reacts in the same way?

To analyze this possibility I decided to take 3 node randomly from each hierarchical level of an Internet-like graph of 1000 nodes, the number of levels on this graph was 4. The total number of destinations was 12 and for each one of them I executed an experiment with multiple MRAI strategies and multiple possible MRAI values.

The properties of this environment are summarized in Table 5.4.

Given that we are evaluating the impact of nodes by their distance from the center of the network, it could be a good way also to test strategies which goal is to enforce those points, for this reason I chose those two strategies.

The reverse of the DPC strategy is simply to opposite of it, it is going to set a higher MRAI value for those nodes that are in the first part of the propagation graph.

Property	Value
Seeds	$[1, 10]$
Signaling	"AWAWAWAW"
Withdraws delay	Uniform distribution between 0.1 s and 5 s
Announcement delay	Uniform distribution between 0.1 s and 5 s
Link delay	Uniform distribution between 0.0001 s and 0.5 s
MRAI	$[0, 60]$ with steps of 10
Number of levels	4
Random dst per level	3
MRAI strategies	DPC, reverse DPC

Table 5.4: Hierarchical experiments environment properties

6 RFD and MRAI correlation

RFD is another parameter of BGP used to avoid messages storms. It is used to avoid flapping routes to continuously make the network unstable. When a network flaps a certain value is increased and when it overpass a threshold then the route is suppressed and not advertised anymore until it goes back below the threshold (or after a certain time).

RFD, other than MRAI, is one of the most studied parameters of BGP because of its influence in the convergence time [11, 12]. RFD received different updates from its first implementation, but recent studies showed that most of the providers still use outdated parameters [10].

The use of deprecated values can lead to a heavy restrictive suppression of small routes delaying the correct spreading of information. Some cases of suppression are caused by faulty interfaces that heavily flaps hundreds of times, while other times is just an update of the node configuration that cause the route to flaps a couple of times and still be suppressed.

In the following chapters, I am going to show how legacy RFD can affect small flaps and how would the new version of RFD react to them. Finally, I would look forward to understand the correlation between RFD and MRAI. When a suppressed route is shared again it could provoke messages storms that triggers different MRAI session, or the opposite case, a low MRAI that cause the growth of the figure of merit that suppresses a route.

6.1 RFD on toy topologies

I firstly studied RFD on toy topologies, to see the effects of it in small networks, like I did in Section 5.1. As a graph, I used a clique of dimension 10, the source of the signalling is connected to the node 0 while the node 5 act as unique servicer for the node x. The node 5 won't be able to share information to node x because of RFD. Node x would have to wait until the route fell below the reuse threshold of node 5 to converge.

The parameters used for RFD are the default *CISCO* parameters, showed in table Table 6.1

Parameter	Value
withdrawal penalty	1.0
re-advertisement penalty	0.0
attribute change penalty	1.0
suppress threshold	2.0
half-life (min)	15 (900s)
Reuse Threshold	0.75
Max Suppress Time (min.)	60 (3600s)

Table 6.1: Cisco default RFD parameters

The parameters of the environment are in Table 6.2
Messages in the signal are delayed by 300 s for two reasons:

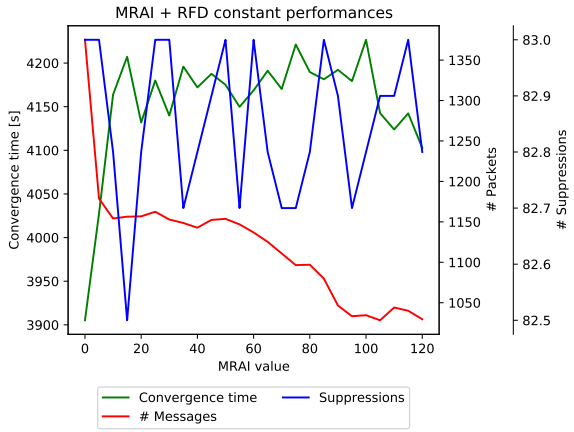
- The main goal of these experiments is to study the correlation from RFD to MRAI and we don't want that MRAI compress parts of the signal.
- I'm trying to simulate one of the possible behaviour tat triggers RFD suppressions, the human faulty reconfiguration of the node.

The signals contain 3 flaps in it, the first one is hipotetically attributed to a configuration that doesn't work properly, the second one is caused by a buggy correction of the configuration and the last one by the introduction of a correct configuration.

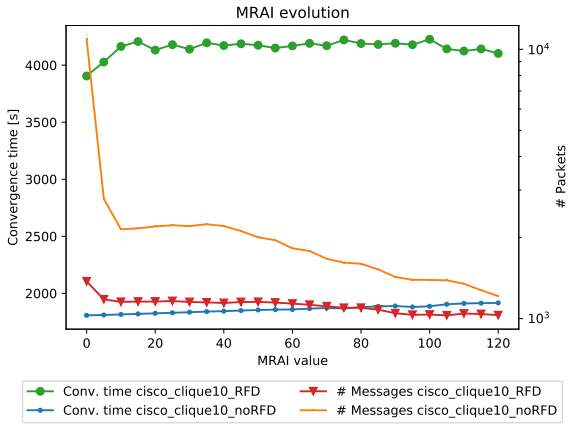
Property	Value
Seeds	$[1, 10]$
Signaling	"AWAWAWA"
Withdraws delay	Constant distribution of 300 s
Announcement delay	constant distribution of 300 s
MRAI	$[0, 120]$
Link delay	Uniform distribution between 0.012 s and 3 s

Table 6.2: Environment parameters used for the experiments on RFD with the clique graph

The MRAI strategy used in all the experiments is the *fixed* one.



(a) Network performances with the standard cisco RFD FiXme: redo this plot with a broader range of suppressions $[81 - 84]$



(b) Network performances standard RFD vs no RFD

Figure 6.1: Evolution of the performances changing MRAI in the links standard RFD vs no RFD, graph clique of 10 nodes, MRAI strategy fixed, signal "AWAWAWA"

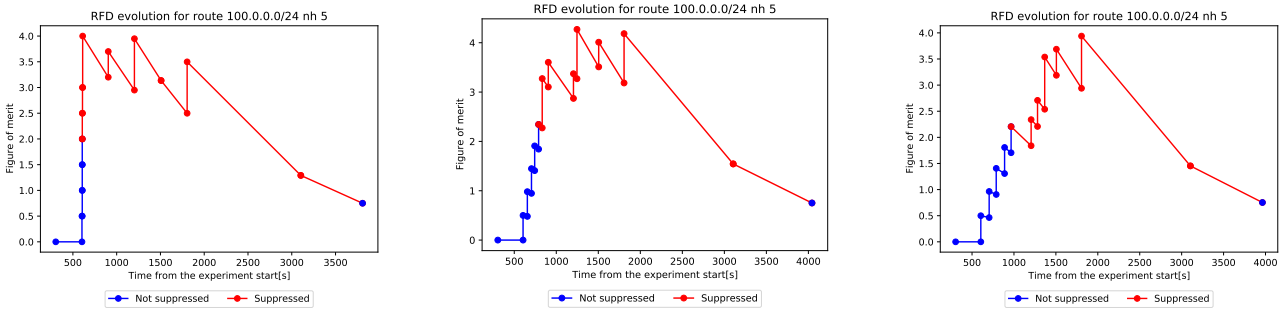
The plot in Figure 6.1a contains a third line that represent the average total number of suppressions detected on the experiment, for each experiment has been executed 10 different runs. The blue line that represents the number of suppressions refers to the third y-axis on the right.

In Fig. 6.1a is possible to see that small changes to MRAI can lead to some small differences in the number of suppressions. Also, the number of messages decreases rapidly and reaches a constant value around 1000, as expected by the passage from an MRAI of 0 s to a few seconds. The nodes that don't trigger a suppression seems to affect also the FiXme: Explain better the next phrase The convergence time stais stable around 4000 s due to the fact that there are almost the same number of suppressions that takes the same ammount of time to be solved

In Figure 6.1b is possible to see the gap between the use of RFD and without it. Notice that the packet axis is in log scale. The difference in the convergence time is due to the fact that with RFD some nodes block the best path that takes a lot of time to become available again. While MRAI grows the set of nodes that suppress routes decreases but the convergence time is highly affected by a restrict subset of them. In our case, for example, the suppressions on nodes 0 and 5 plays an important role. The first one for the spreading in the whole network, the second one for the transmission of information to node x.

For this reason, we can look more deeply on what happened to the figure of merit of node x and five in Figures 6.2 and 6.3

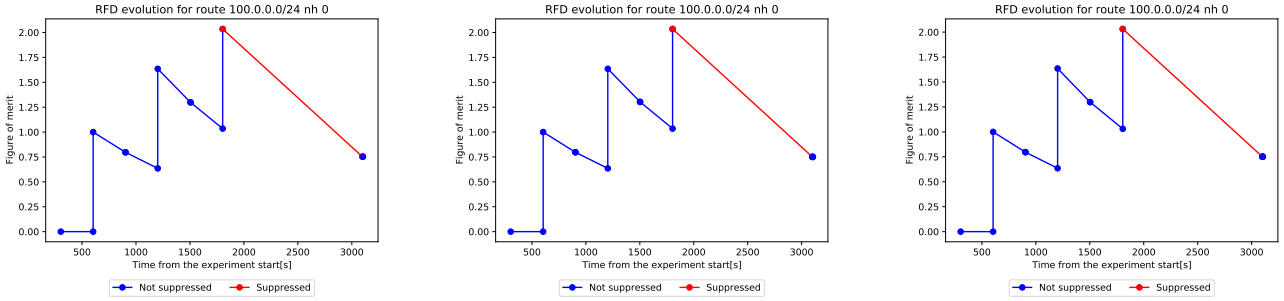
The node x is a leaf of the network that will absorb everything the node 5 sends to it. In Figure 6.2 is possible to see the evolution of the figure of merit with different MRAI values. In the first case, with an MRAI equal to 0 s, we will see a huge spike caused by a lot of messages and route changes that the node 5 sends to it. while in the other two cases Figures 6.2b and 6.2c the MRAI seems to not be much



(a) MRAI = 0s (b) MRAI = 50s (c) MRAI = 100s

Figure 6.2: Evolution of the figure of merit in the node X with different MRAIs

effective on the route through node 5. The messages are more delayed with high MRAI but the growth of the figure of merit has the same trend. In those cases, we can see that the route has been suppressed around 1000 s and is going to become useful again around 4000 s. In this period of time from 1000 s to 4000 s, node x still receives some updates from node 5 that affects its best path, and this makes the figure of merit evolve. The evolution of the figure of merit stops around 2000 s that's because also the node 5 has suppressed the route, Figure 6.3, and doesn't send any more advertisements. The point around 3000 s represent the moment when the route becomes available again for node 5 that communicates the change to x.



(a) MRAI = 0s (b) MRAI = 50s (c) MRAI = 100s

Figure 6.3: Evolution of the figure of merit in the node X with different MRAIs

The evolution of the figure of merit of the best path of node 5 is different from the one of node x. In fact, it is not influenced by MRAI as we can see in Figure 6.3. That because the node 5 is directly connected to the node 0 that every 300 s forward the message of d. 300 s are a too large delay to be affected by the compression effect of MRAI. Around 2000 s node 5 suppress the route (as any other node in the clique) and stops to forward it to node x until 3000 s when it becomes available again.

Node x took almost 4000 s to converge because of the big fluctuations of node 5 that suffers of the *Path Exploration* problem, path changes are considered bad behaviour in RFD.

In conclusion, we can say that RFD can be affected by MRAI and that RFD can prevent a lot of messages at the cost of very high convergence times.

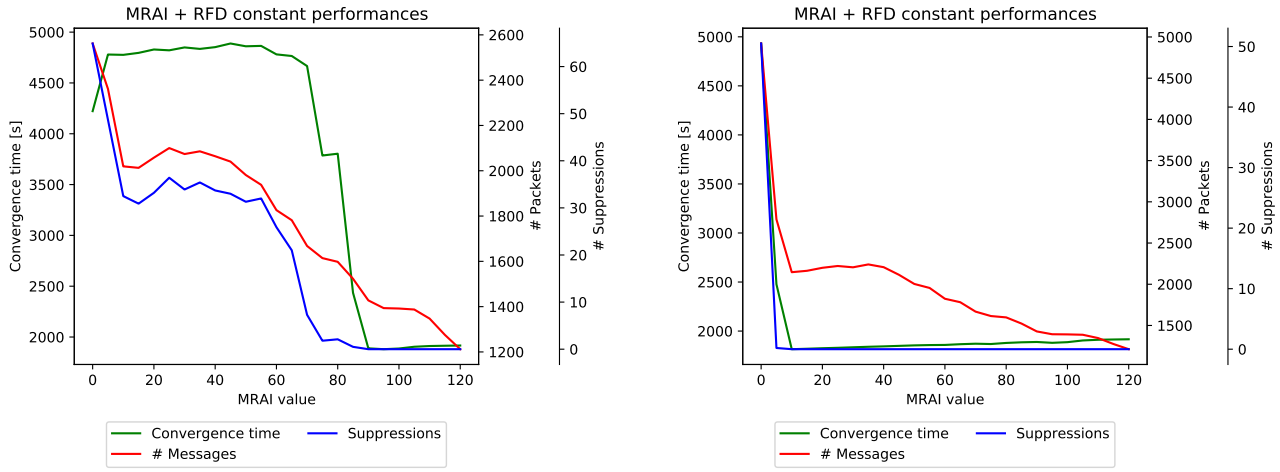
6.2 RFC 2439 VS RFC 7196

The difference in the two RFC that defines RFD [3, 15] is in the parameters used. Infact the last RFC introduce two new set of parameters where the figure of merit threshold is increased up to at least 6.0. The two categories are:

- ***Aggressive***, Suppression threshold no less than 6.0;
- ***Conservative***, Suppression threshold no less than 12.0;

Respectively 3 and 6 times the actual standard.

I have then repeated the same experiments of Section 6.1 with the same clique graph, but with the two new RFD strategies, the results are showed in Figure 6.4.



(a) RFD 7196 Aggressive on the clique topology

(b) RFD 7196 Conservative on the clique topology

Figure 6.4: MRAI influence with different RFD strategies from [15]

We can see two completely different evolutions of the performances in Figure 6.4. On the left plot, we can see the evolution with the *Aggressive* strategy. and MRAI is more effective to this strategy in respect of the standard one. The number of suppressions fell down to almost 0 with an MRAI near 90 s. The message trend is similar to the one of the case without RFD but with an important difference in the case of MRAI equal 0 s, the number of average messages is around 2600 in respect of the 10 000 without RFD. While with a high MRAI the message trends are similar and equal when the number of suppressions reaches 0.

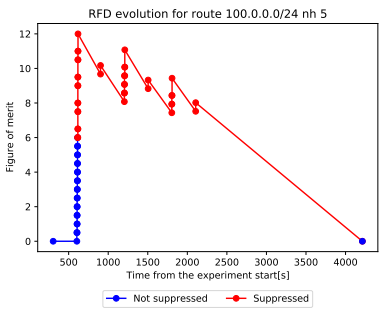
The convergence time, on the other hand, has a differen trend in respect of the one that we saw in Figure 6.1a. Here we see a descending trend caused by the fact that MRAI is able to avoid some messages and, as a consequence, avoid the growing of the figure of merit in some nodes permitting to the convergence time to decrease. Once the number of suppressions reaches 0 obviously the network performances are equal as in the *NoRFD* case.

In Figure 6.4b we can see the evolution of the network with the *Conservative* strategy, the threshold of this strategy is the double of the *Aggressive* strategy. The effects of this difference are huge, is sufficient an MRAI of 10 s to avoid at all suppressions, causing the trend, in terms of messages and convergence time to be equal to the no RFD case. Also with an MRAI of 0 s is possible to see a difference in terms of messages and convergence time in respect of the other two strategies. This is the strategy that more likely resembles the *NoRFD* one, having a convergence time incredibly more low, at the cost of few hundreds messages.

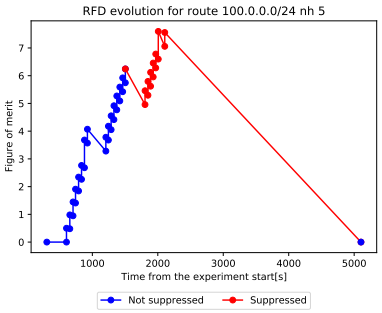
We can now take a look more closely to what happens to the figure of merit for the only route that node x receives. Results are exposed in Figure 6.5

We can see in Figures 6.5a and 6.5b that MRAI plays an important role in the figure of merit of node x. In the first case, the route would be delayed up to 4000 s with a threshold that touches 12 few seconds after the first flap. In the second one, the growth is slower but it passes the threshold around 1800 s reaching a value of 7. For this reason it requires a higher time to become available again. We can also notice that the route become available with a figure of merit of 0, thats because it has been triggered the max suppression threshold, whith the defaul value of cisco after 1 h a route would become available again, no matther the evlution of the figure of merit. With a higher MRAI node 5 is able to compress more routes, the effects are visible in Figure 6.5c, where the figure of merit never goes over the threshold.

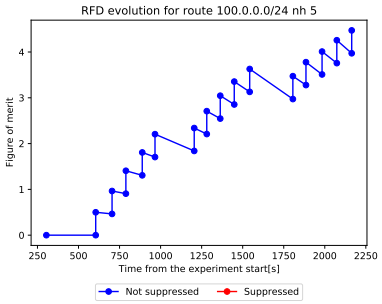
In conclusion, if before MRAI, with the standard RFD was playing a more marginal role because of the restrictive threshold, now, with those strategies it plays a more relevant role and act as a key factor between the suppression or not.



(a) MRAI = 0s, RFD 7196 Aggressive, clique topology



(b) MRAI = 50s, RFD 7196 Aggressive, clique topology

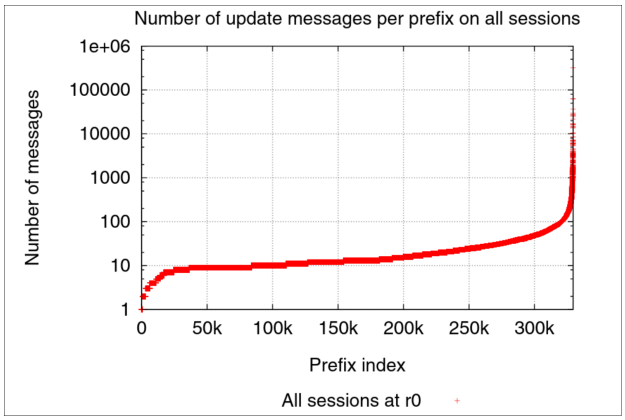


(c) MRAI = 100s, RFD 7196 Aggressive, clique topology

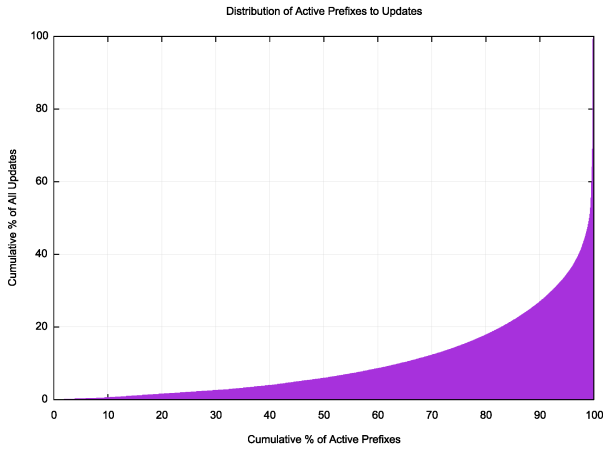
Figure 6.5: Evolution of the figure of merit in the node X with different MRAIs, with RFD 7196 aggressive in a clique topology

6.3 Mice VS Elephants

From the work of R. Bush et al., [11] we know that the majority of updates that are transmitted on the Internet are from a small set of AS. Those ASes with their flaps causes update storms almost continuously. I report a figure from [11] for simplicity in Figure 6.6a Thanks to the studies of APNIC FiXme: insert citation of footnote, something we also know that this behaviour is still present nowadays, the Figure 6.6b is taken from one of their annual reports and shows that the 10% of all the active prefixes produce more or less the 70% of the total updates.



(a) Prefixes and number of updates associated, figure from [11]



(b) Prefixes and number of updates associated, [apnic 2019]

Figure 6.6: Prefixes influence on updates

We can then divide those prefixes in two sets:

- ***Mice***, this set represent the majority of the prefixes, all the prefixes that does not generate more than 100 updates in Figure 6.6a
- ***Elephants***, this set represent the remaining part of the prefixes, those that produces the majority of the messages.

Thanks to a review of a BGP year by APNIC, presented at RIPE 52 [27], we can also have an example of those elephants prefixes. This example is shown in Figure 6.7, it takes in consideration the prefix "202.64.49.0/24" showing that in a relatively small period of time it has produced thousands of ADV per day. In this case, this particular prefix has produced 198.370 ADV producing in total 96.330 flaps.

I have then used this data to configure two new environments for the simulations. The first one that points to reproduce the *Mice* behaviour, the second one the *Elephants*.

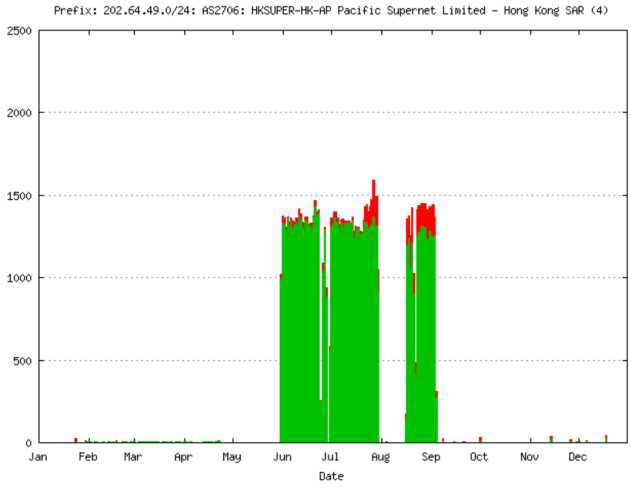


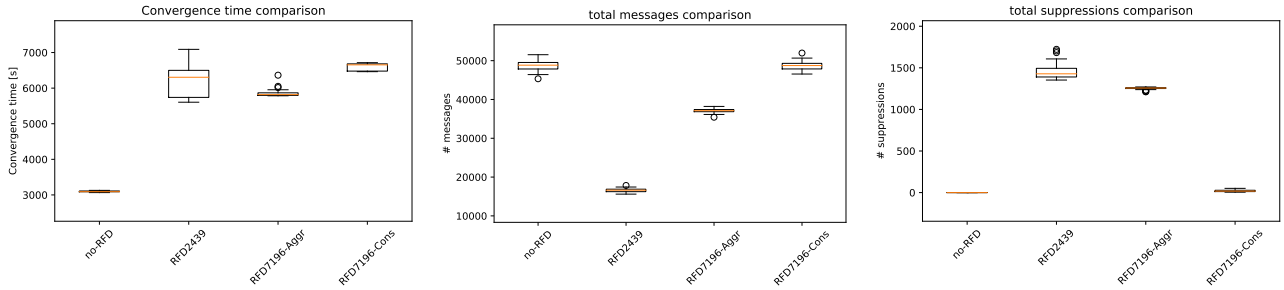
Figure 6.7: 202.64.49.0/24 flaps plot from [27]

In both these environments, I have then compared the four different strategies of RFD, *NoRFD*, standard RFD and the two from [15].

The topology used for those experiments is an *Internet like* topology with 1000 nodes and MRAI is fixed to 30 s for all the links. The source of the signal has been chosen randomly on the graph. For each experiment has been then executed 50 runs.

6.3.1 Mice

The particularity of the *Mice* experiments is in the signal, we have a low number of flaps interleaved by a long timer. I have then used a signal with 5 flaps, "AWAWAWAWAWA" with a delay of 300 s (5 min) between each message. The results are presented in Figure A.8. I have executed 50 runs for each RFD strategy.



(a) Convergence time respect to the RFD strategy

(b) Number of messages respect to the RFD strategy

(c) Number of suppressions respect to the RFD strategy

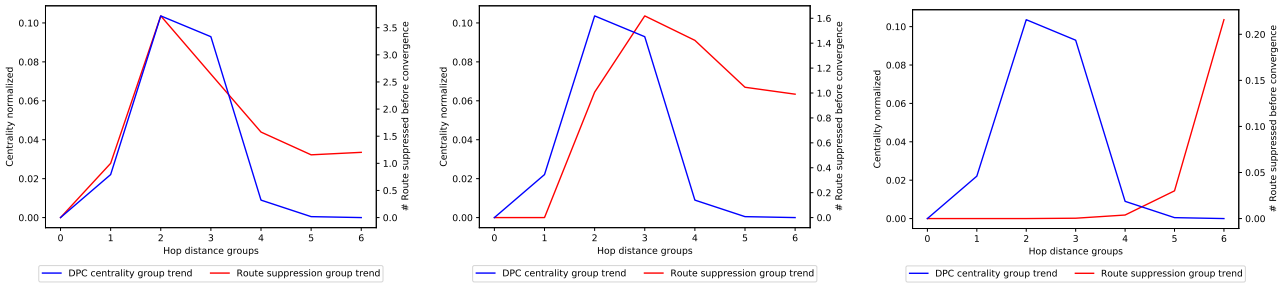
Figure 6.8: Internet like topology 1000 nodes, MRAI=30s, random destination, 5 flaps, 300 s message delay, Network performances, 50 runs per strategy.

From Figure 6.8c we can see that there is a big difference in terms of suppressions. The standard strategy produces on average almost 1500 suppressions and the effects of those suppressions can be seen in Figures 6.8a and 6.8b because, on average, it presents a convergence time higher than 6000 s but with a number of total messages transmitted around 16 000 with a very low varaiance. A different case is presented by the *Conservative* strategy from RFC 7196 [15]. The threshold in this last case is so permessive that we have a really small number of suppression. For this reason the number of messages transmitted, on average, is really similar to the *NoRFD* case, around 50 000. While, the convergence time is around 6500 s, like the standard RFD strategy. This is prove that few suppression can heavily influence the network performances, in particulare the convergence time. In the middle we find the *Aggressive* strategy, we can see from the suppression boxplot that it produces a smaller number of suppressions in respect of the legacy strategy with a smaller varaiance. Also the nconvergence time respect this trend, infact, the average time is below 6000 s. While, The number of messages transmitted

is more than the double in respect of the strategy described by the RFC 2439.

We can than conclude that a small number of suppression can affect both the performances, like the few suppressions in the *Conservative* strategy for the convergence time. Also the few missing suppression in the *Aggressive* strategy will enormously impact the number of messages transmitted.

We can then study which are the nodes that produce the suppressions and how far are them from the signal source. We can see the results of this study, for each suppression technique in Figure 6.9.



(a) RFD 2439 Strategy (b) RFD 7196 Aggressive Strategy (c) RFD 7196 Conservative Strategy

Figure 6.9: Internet like topology 1000 nodes, MRAI = 30 s, random destination, 5 flaps, 300 s between messages, Suppression trend VS avg hop centrality

FiXme: Adjust the y-axis

For the plots in Figure 6.9 the x axis represent the distance from the source node in terms of hops and all the other nodes are grouped by this distance. The blue line represents the average centrality of the groups, for each node of the graph I calculated the centrality using the DPC metric then grouped them and calculated the average value. As expected the central nodes have a higher centrality and them are at few hops of distance from the source node. The centrality trend is equal for each plot in Figure 6.9 because the graph and the source node are the same for each experiment.

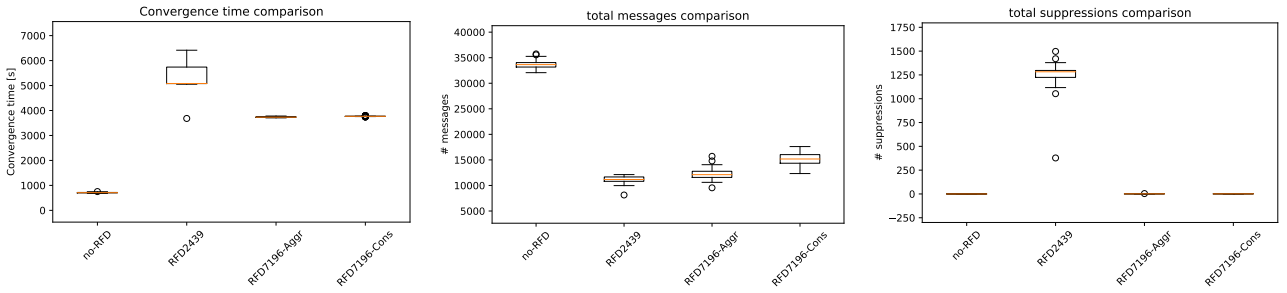
The red line represents the average number of suppressions per group. As we can see with the standard strategy, Figure 6.9a, on average, the route is blocked 1 time by the nearest nodes and then, this value increase reaching the center clique up to 3.5 times and then slowly decreases in the following groups. In the farest group we will still see, on average 1 suppression. The *Aggressive* strategy, Figure 6.9b present a similar behaviour, the nearest nodes don't blocks the route, while the central nodes starts blocking it with a maximum average of 1.6 times. Aftter those central nodes, the farest nodes, that have a low entrality will block it on average 1 time, like the legacy strategy. The *Conservative* strategy, presented in Figure 6.9c, has a different trend. We can see that the central nodes does not block the route, while only the farest ones blocks it few times, with an average value of 0.2 times. This can gives us some hints, a very high threshold can promote the path exploration problem that will cause multiple update storms in farest nodes.

FiXme: Add a conclusion

6.3.2 Elephants

The elephants prefixes, as I mentioned in Section 6.3, are the ones that produce the majority of the ADV. And we also know, thanks to [27], that is possible to see over thousands of messages per day. For this reason, the *elephants* environment signal is composed by 100 flaps, with a delay between the messages of 3 s. All the other properties of the environment are unchanged. The results are presented in Figures 6.10 and 6.11.

Is possible to see in Figure 6.10 that this time we have a different behaviour from all the 3 RFD strategies. In Figure 6.10c we can see that the The standard strategy does more than 1250 suppression on average, producing the lowest number of messages, around 11 000 but the highest convergence time with more than 5000 s. All the suppression are trigger by the *Path Exploration* problem that causes repitetvly ADV storms that trigger the suppressions on the majority of the nodes. The two new strategies would produce on average just few suppressions in respect of the legacy one, but the number of messages doesn't differ too much. While there is a huge improvement on the convergence time, on average, both the new strategy permits to the network to converge in less than 4000 s. All the three



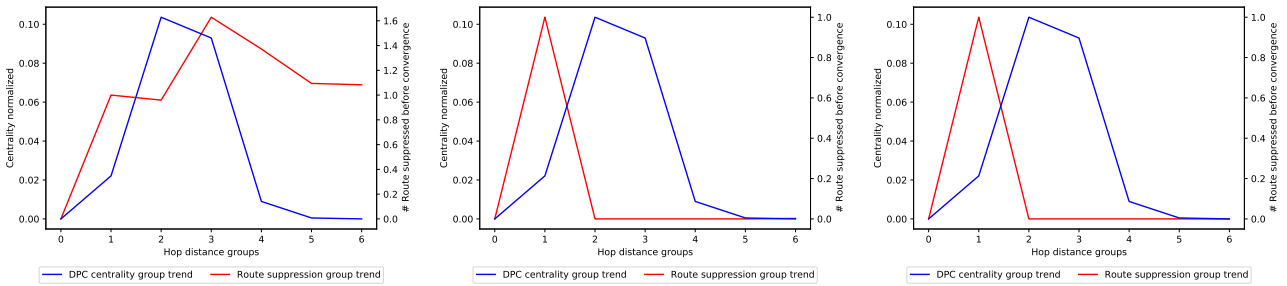
(a) Convergence time respect to the RFD strategy

(b) Number of messages respect to the RFD strategy

(c) Number of suppressions respect to the RFD strategy

Figure 6.10: Internet like topology 1000 nodes, MRAI = 30 s, random destination, 100 flaps, 3 s delay, Network performances

strategy produce 1/3 of the messages produced by the *NoRFD* strategy.



(a) RFD 2439 Strategy

(b) RFD 7196 Aggressive Strategy

(c) RFD 7196 Conservative Strategy

Figure 6.11: Internet like topology 1000 nodes, MRAI = 30 s, random destination, 100 flaps, 3 s delay, suppressions by distance from the source

We can see in Figure 6.11 the comparison between the average number of suppressions per node group of the different strategies. In Figures 6.11b and 6.11c we can notice that both strategies reacts in the exact same way at the elephant environment. The only nodes that suppress the route are the nodes that are closer to the source. All the other nodes of the network doesn't experience enough messages to block the route. In the first figure, Figure 6.11a, we can see that, on average, every node suppress at least one time the source of the signal. The hipothesis behind this trend is that the intervention of the closer nodes is not timely enough and all the other nodes have the time to experience the path exploration problem. With a low threshold is sufficient a small number of ADV storms caused by the *Path Exploration* to trigger the RFD suppression.

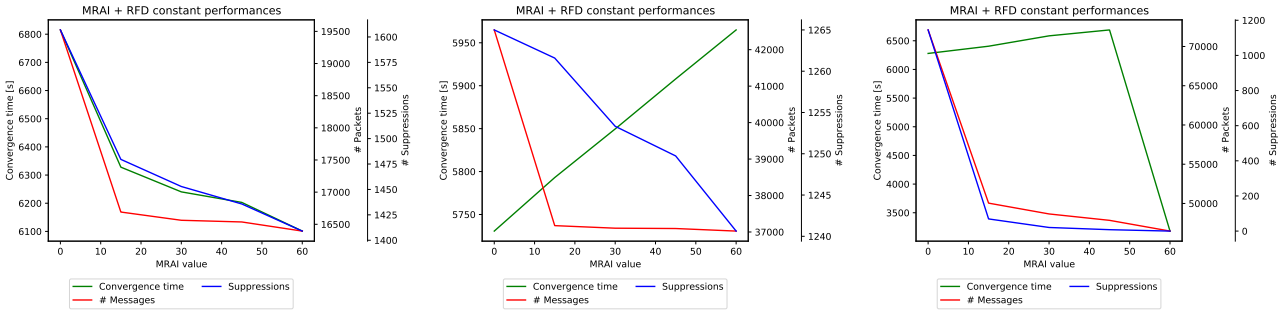
FiXme: Ri-elaborate the end of the subsection

6.3.3 MRAI influence on Mice and Elephants

We can now study the influence of MRAI on those two cases. The environments are equal to the previous section. The results of the *Mice* case are exposed in Figure 6.12, while the results of the elephant case are in Figure 6.13.

FiXme: Redo this graphs with more MRAI values, update the figures with the same y-range

We can see in Figure 6.12 how the different RFD strategies reacts, on the same topology, whith different MRAI settings. In Figure 6.12a are presented the network performances with the legacy RFD strategy from the RFC 2439 [3]. First of all we can see the influence of MRAI on the number of suppressions that decrease from 1600 with an MRAI equal to 0 s to almost 1400 with MRAI = 60 s. We can notice that the message trend reacts as expected with the increasing of MRAI, but its noticeable that with an MRAI of 0 s there are less than 20 000 messages thanks to RFD. The convergence time doesn't have the same trend as other MRAI experiments, it has a decreasing trend, while we were expecting an increasing one. This is caused by the routes that doesn't suppress anymore the route. It is greater the gain obtained by the suppression reduction than the disadvantage caused by MRAI that



(a) RFD 2439 Strategy (b) RFD 7196 Aggressive Strategy (c) RFD 7196 Conservative Strategy

Figure 6.12: Internet like topology 1000 nodes, random destination, 5 flaps, 300 s delay, Network performances, MRAI strategy fixed

requires to the node to wait mroe time.

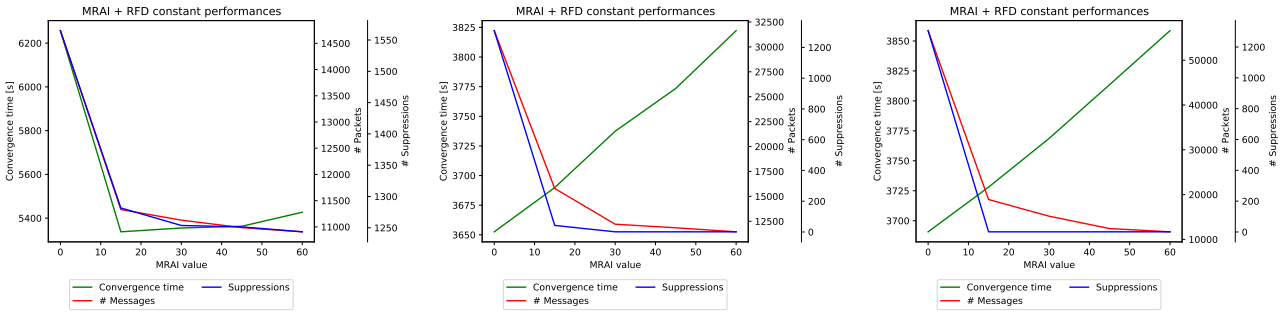
The second case we can analyze is the *Aggressive* strategy presented in Figure 6.12b. We can easily notice that this time the variation in terms of suppression is smaller, going from a value of 1265 to 1240 suppressions. The number of messages has a similar trend to the standard strategy, but whith a different range. At the beginning, with MRAI = 0 s there are more than 42 000 messages, after a while it converges around a value of 37 000 messages. The convergence time has the expected trend by the growth of MRAI. The number of messages higher is caused by the fact that RFD requires more time to activate itself, and in the meanwhile a lot of messages storm will pass the network. The convergence time, in this case is not affected by the decrease of the number of suppressions. The gain obtained by the RFD suppressions trend doesn't compensate the effects of MRAI that makes the convergence time grow.

In the last strategy, the *Conservative* one, we can see an other, different behaviour. In terms of suppressions MRAI makes a huge difference, we go from 1200 suppressions to 0 and just an MRAI of 15 s reduce it around 50. Also in this case the number of messages are higher than the other two strategies. Like before for the *Aggressive* strategy this is caused by the not restrictive thresholds. An interesting behaviour can be seen for the convergence time, infact like for the *Aggressive* strategy it starts grwoing also with more than 1000 suppressions of difference, but as soon the number of suppression tuches 0 it goes back to the *NoRFD* behaviour.

All this comparison can be also seen in Figures A.8 and A.9.

Figure 6.12a compared with Figure 6.12b tels us that not all the gain in terms of suppressions gives also advantages in termes con convergence time. FiXme: Not sure of this phrase, it could be due to the threshold. Also a difference of few hundred suppression have a huge impact in the messages transitted.

In Figure 6.13 are presented the results obtained with the elephant environment and multiple MRAI values.



(a) RFD 2439 Strategy (b) RFD 7196 Aggressive Strategy (c) RFD 7196 Conservative Strategy

Figure 6.13: Internet like topology 1000 nodes, random destination, 100 flaps, 3 s delay, Network performances

The trends in the elephant case are compleatly different in respect of the mice environment. Starting from the standard strategy in Figure 6.12a we can see that the number of suppressions decreases of few

hundred units thanks to a higher MRAI. Also the number of messages decrease from around 14 500 reaching a stable state around 11 000. While the convergence time benefits of the suppression rate decrease reaching a valley around 5400. But, after that point the effect of the next suppressions is not enough to keep a descending trend, while MRAI acquire a more predominant position making the convergence time slightly increasing. A different behaviour can be saw in Figures 6.13b and 6.13c, where, the number off suppressions thanks to MRAI reaches a number slightly higher than 0. The number of messages reaches the same convergence point around 12 500 but with a compleatly different starting point. with MRAI at 0 s the aggressive strategy present a number of messages around 32 500 while the conservative strategy is around 60 000, almost the double of the *Aggressive* one. This huge difference is caused by the fact that the conservative strategy requires more flaps to overcome the suppressioon threshold, and all those messages can cause a more and more updates storms, due to the *Path exploration* problem in the other parts of the network. In both, *Aggressive* and *Conservative* strategy the convergence time is not affected by the variation on the number of suppressions but it's only affected by the growth of MRAI.

The fact that in the last two strategies the time is not affected by the huge number of suppressions difference could be saw as an error, but it is not. In fact, the suppressions with an MRAI of 0 s happens few seconds after the beginning of the experiment and the majority of the nodes will suppress the route, but we know that after 3600 s a route can't be suppressed anymore. For this reason after that time, there will be just a last update storm to propagate the reintroduction of the route (this will cause some suppressions too). and on average all the nodes will converge few seconds after 1 h

We can then conclude that MRAI influences both the *Mice* and *Elephants* cases. The major effects can be saw on the two modern strategies of RFD. For the *Mice* environment those two strategies will tend to have a behaviour similar the the *NoRFD* strategy. and MRAI would influence the number of suppressions and indirectly the convergence time and the number of messages transmitted. We can see from Figure A.9 that also the set of nodes affect by suppressions changes. We can see even more effects in the *Elephants* environment, where MRAI would affect the number of suppressions of both the *Aggressive* and *Conservative* strategy that would keep just few suppression in comparison of the thousands of suppressions with the legacy 2439 strategies. And the new strategies would have a highly impact on the convergence time, at the cost of few hundred messages on average. In Figure A.11 is possible to see the effects on the set of nodes that effectively suppress the route, in the legacy case even the more distance nodes would suppress it, while with the new strategies is sufficient a suppression near the source, and MRAI would help to prevent suppressions due to the *Path exploration* problem.

7 Conclusion

- Wrap up
- Path exploration explosion of the FSM
- MRAI convergence dependency
- RFD and MRAI co-dependency

7.1 Future Works

:)

Bibliography

[1] Y. Rekhter, T. Li, and S. Hares, "A Border Gateway Protocol 4 (BGP-4)," RFC 4271, Internet Engineering Task Force, Tech. Rep. 4271, Jan. 2006, updated by RFCs 6286, 6608, 6793, 7606, 7607, 7705.

[2] M. L. Daggitt and T. G. Griffin, "Rate of convergence of increasing path-vector routing protocols," in *2018 IEEE 26th International Conference on Network Protocols (ICNP)*. IEEE, 2018, pp. 335–345.

[3] C. Villamizar, R. Chandra, and R. Govindan, "Bgp route flap damping," RFC 2439, Tech. Rep., 1998.

[4] A. Fabrikant, U. Syed, and J. Rexford, "There's something about mrai: Timing diversity can exponentially worsen bgp convergence," in *2011 Proceedings IEEE INFOCOM*. IEEE, 2011, pp. 2975–2983.

[5] T. G. Griffin and B. J. Premore, "An experimental analysis of bgp convergence time," in *Proceedings Ninth International Conference on Network Protocols. ICNP 2001.* IEEE, 2001, pp. 53–61.

[6] P. Jakma, "Revised default values for the bgp'minimum route advertisement interval'," *draft-jakma-mrai-02. txt (Internet Draft)*, 2008.

[7] J. Qiu, R. Hao, and X. Li, "The optimal rate-limiting timer of bgp for routing convergence," *IEICE Transactions on Communications*, vol. 88, no. 4, pp. 1338–1346, 2005.

[8] P. Jakma, "Revisions to the bgp'minimum route advertisement interval'," *Internet Draft draft-ietf-idr-mrai-dep-02*, 2010.

[9] M. Milani, "BGP e Load Centrality: Implementazione del calcolo della centralità nel protocollo BGP." [Online]. Available: http://dit.unitn.it/locigno/preprints/Milani_Mattia_laurea_2017_2018.pdf

[10] C. Gray, C. Mosig, R. Bush, C. Pelsser, M. Roughan, T. C. Schmidt, and M. Wahlisch, "Bgp beacons, network tomography, and bayesian computation to locate route flap damping," in *Proceedings of the ACM Internet Measurement Conference*, 2020, pp. 492–505.

[11] C. Pelsser, O. Maennel, P. Mohapatra, R. Bush, and K. Patel, "Route flap damping made usable," in *International Conference on Passive and Active Network Measurement*. Springer, 2011, pp. 143–152.

[12] Z. M. Mao, R. Govindan, G. Varghese, and R. H. Katz, "Route flap damping exacerbates internet routing convergence," in *Proceedings of the 2002 conference on Applications, technologies, architectures, and protocols for computer communications*, 2002, pp. 221–233.

[13] P. Smith and C. Panigl, "Ripe routing working group recommendations on route-flap damping," *ripe-378, May*, 2006.

[14] R. Bush, C. Pelsser, M. Kuhne, O. Maennel, and K. E. R. Mohapatra, P.and Patel, "Ripe routing working group recommendations on route-flap damping," *ripe-580, January*, 2013.

[15] C. Pelsser, R. Bush, K. Patel, P. Mohapatra, and O. Maennel, "Making route flap damping usable," Tech. Rep., 2014.

[16] N. Matloff, "Introduction to discrete-event simulation and the simpy language," *Davis, CA. Dept of Computer Science. University of California at Davis. Retrieved on August*, vol. 2, no. 2009, pp. 1–33, 2008.

[17] G. Dagkakis, C. Heavey, S. Robin, and J. Perrin, "Manpy: An open-source layer of des manufacturing objects implemented in simpy," in *2013 8th EUROSIM Congress on Modelling and Simulation*. IEEE, 2013, pp. 357–363.

[18] A. Elmokashfi, A. Kvalbein, and C. Dovrolis, "On the scalability of bgp: The role of topology growth," *IEEE Journal on Selected Areas in Communications*, vol. 28, no. 8, pp. 1250–1261, 2010.

[19] T. G. Griffin, "A Finite State Model Update Propagation for Hard-State Path-Vector Protocols."

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

[21] S. Deshpande and B. Sikdar, "On the impact of route processing and mrai timers on bgp convergence times," in *IEEE Global Telecommunications Conference, 2004. GLOBECOM'04.*, vol. 2. IEEE, 2004, pp. 1147–1151.

[22] L. Maccari and R. Lo Cigno, "Improving Routing Convergence With Centrality: Theory and Implementation of Pop-Routing," *IEEE/ACM Trans. on Networking*, vol. 26, no. 5, pp. 2216–2229, Oct. 2018.

[23] L. Maccari, L. Ghiro, A. Guerrieri, A. Montresor, and R. Lo Cigno, "On the Distributed Computation of Load Centrality and Its Application to DV Routing," in *37th IEEE Int. Conf. on Computer Communications (INFOCOM)*, Honolulu, HI, USA, Apr. 2018, pp. 2582–2590.

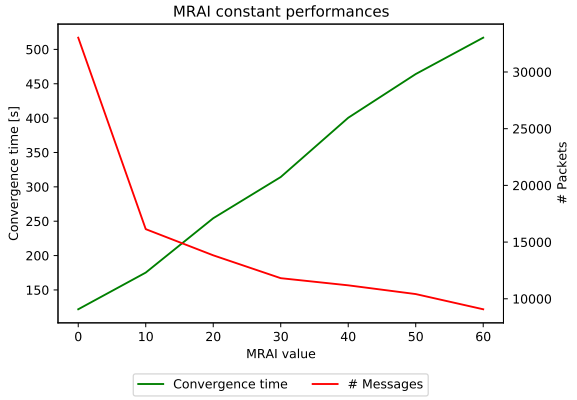
[24] M. Milani, M. Nesler, M. Segata, L. Baldesi, L. Maccari, and R. L. Cign o, "Improving bgp convergence with fed4fire+ experiments," in *IEEE INFOCOM 2020-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 2020, pp. 816–823.

[25] C. Labovitz, A. Ahuja, A. Bose, and F. Jahanian, "Delayed internet routing convergence," *ACM SIGCOMM Computer Communication Review*, vol. 30, no. 4, pp. 175–187, 2000.

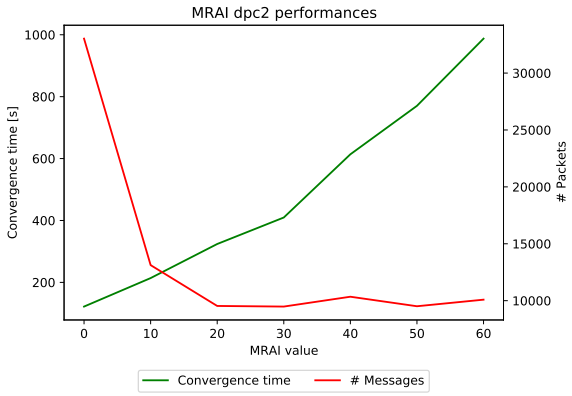
[26] E. Goodarzi, M. Ziaei, and E. Z. Hosseinipour, *Introduction to optimization analysis in hydrosystem Engineering*. Springer, 2014.

[27] G. Huston, "a bgp year in review," 2006. [Online]. Available: https://meetings.ripe.net/ripe-52/presentations/ripe52-plenary-bgp-review.pdf

Appendix A Appendix

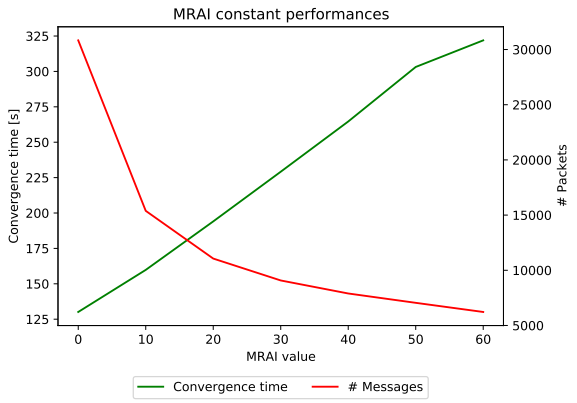


(a) Network perforcances, *fixed* MRAI strategy

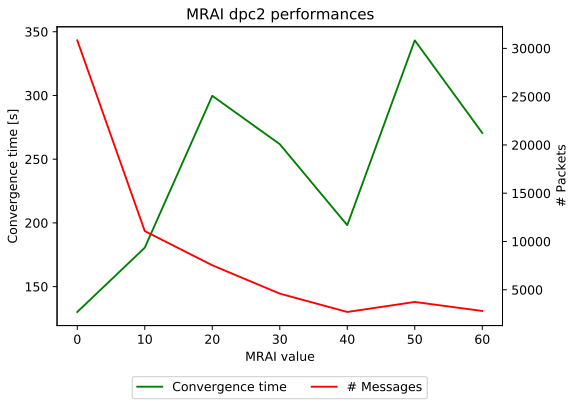


(b) Network perforcances, DPC MRAI strategy

Figure A.1: Network perfomances comparison with different MRAI strategies, Graph internet like with 1000 nodes, signal "AWAW"



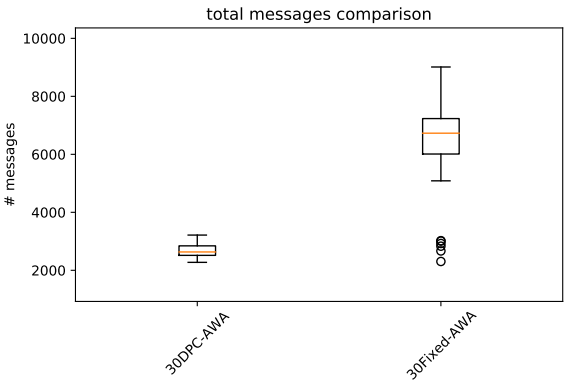
(a) Network perforcances, *fixed* MRAI strategy



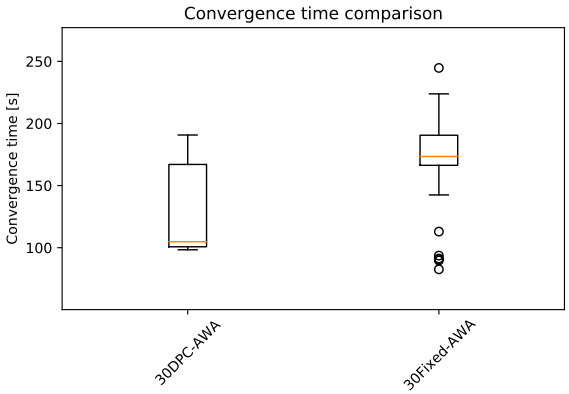
(b) Network perforcances, DPC MRAI strategy

Figure A.2: Network perfomances comparison with different MRAI strategies, Graph internet like with 1000 nodes, signal "AWAWA"

FiXme: Produce again those plots in Figure A.7

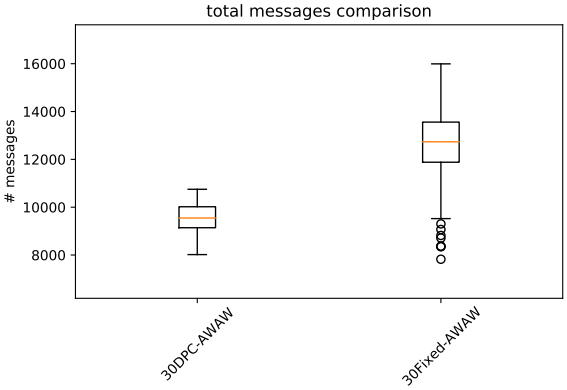


(a) Network perforcances, messages necessary to reach convergence with different MRAI strategies

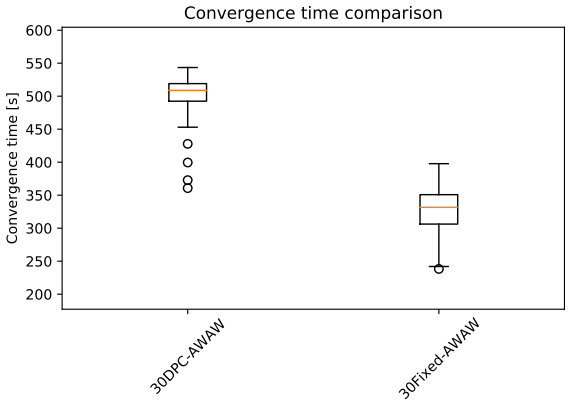


(b) Network perforcances, time required to reach convergence with different MRAI strategies

Figure A.3: Network perfomances comparison with different MRAI strategies, Graph internet like with 1000 nodes, MRAI value 30 s, number of runs for each strategy 100, signal "AWA"

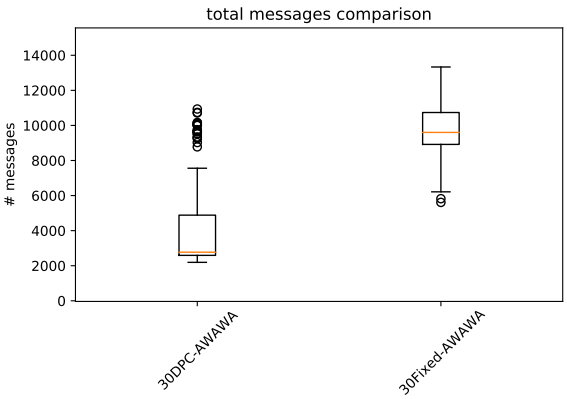


(a) Network perforcances, messages necessary to reach convergence with different MRAI strategies

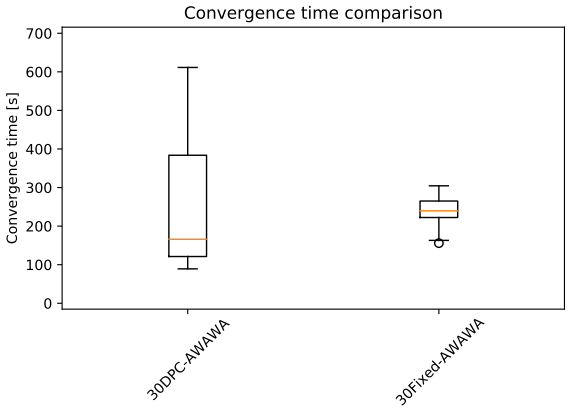


(b) Network perforcances, time required to reach convergence with different MRAI strategies

Figure A.4: Network perfomances comparison with different MRAI strategies, Graph internet like with 1000 nodes, MRAI value 30 s, number of runs for each strategy 100, signal "AWAW"



(a) Network perforcances, messages necessary to reach convergence with different MRAI strategies



(b) Network perforcances, time required to reach convergence with different MRAI strategies

Figure A.5: Network perfomances comparison with different MRAI strategies, Graph internet like with 1000 nodes, MRAI value 30 s, number of runs for each strategy 100, signal "AWAWA"

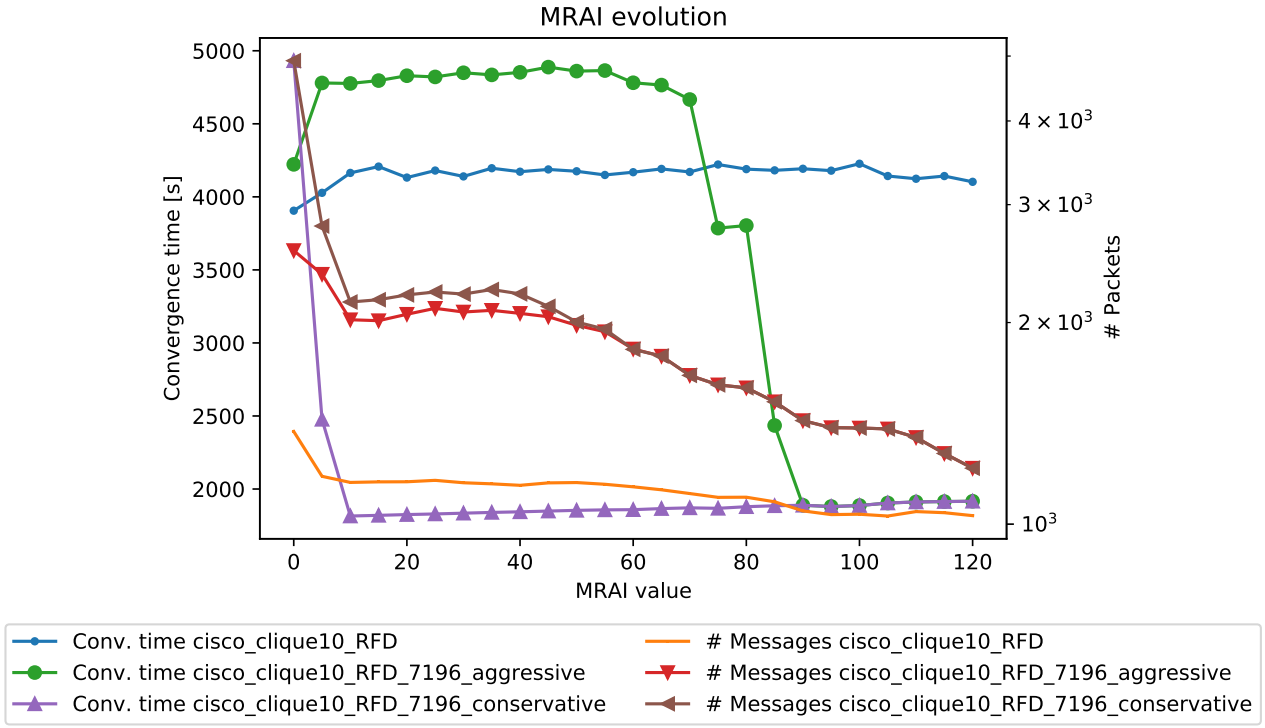
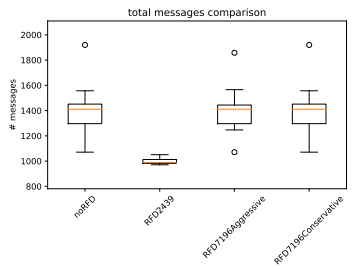
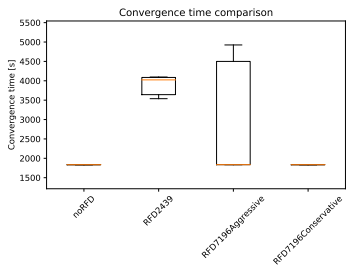


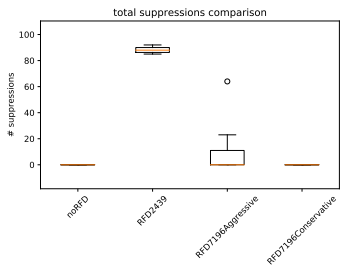
Figure A.6: Comparison of the *clique* topology with RFD 2439 and the with RFD 7196 strategies



(a) clique topology, MRAI=30s, 10 runs, Messages comparison

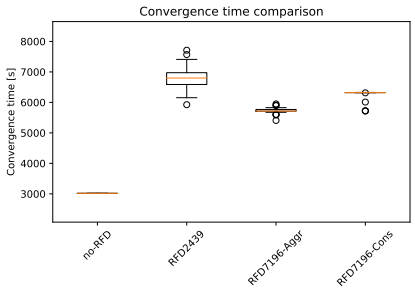


(b) clique topology, MRAI=30s, 10 runs, Convergence time

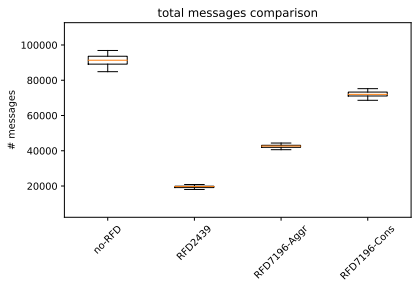


(c) clique topology, MRAI=30s, 10 runs, Number of suppressions

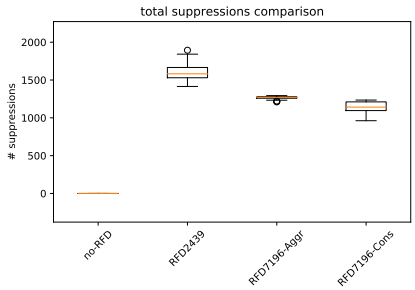
Figure A.7: Clique topology, MRAI=30s, 10 runs, comparison of the network performances



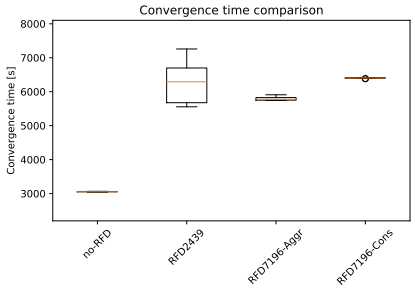
(a) Convergence time respect to the RFD strategy, MRAI=0s



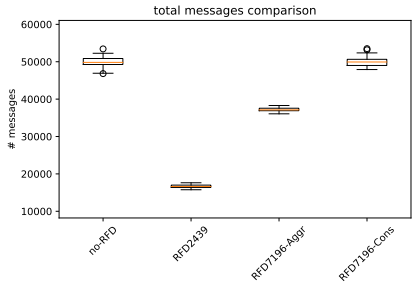
(b) Number of messages respect to the RFD strategy, MRAI=0s



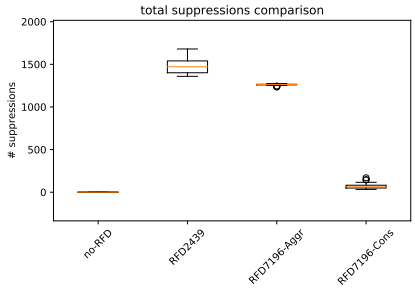
(c) Number of suppressions respect to the RFD strategy, MRAI=0s



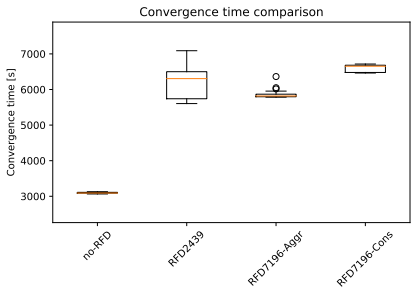
(d) Convergence time respect to the RFD strategy, MRAI=15s



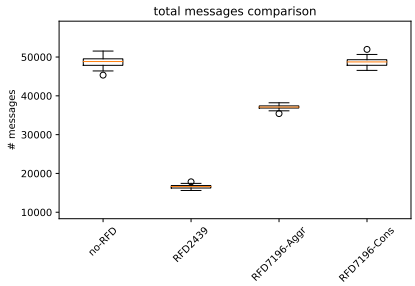
(e) Number of messages respect to the RFD strategy, MRAI=15s



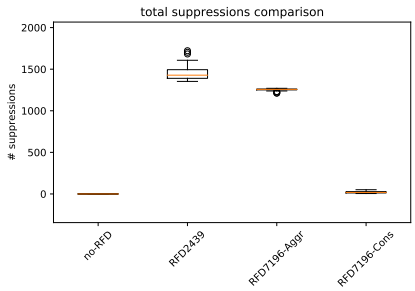
(f) Number of suppressions respect to the RFD strategy, MRAI=15s



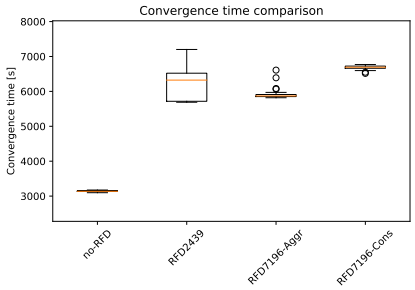
(g) Convergence time respect to the RFD strategy, MRAI=30s



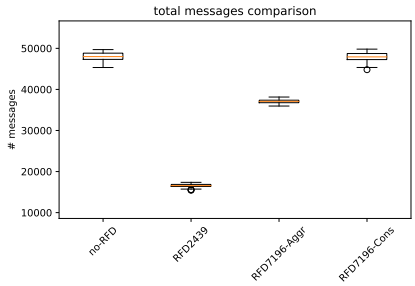
(h) Number of messages respect to the RFD strategy, MRAI=30s



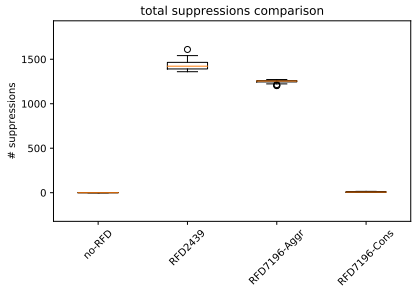
(i) Number of suppressions respect to the RFD strategy, MRAI=30s



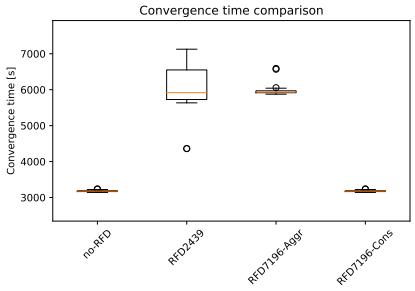
(j) Convergence time respect to the RFD strategy, MRAI=45s



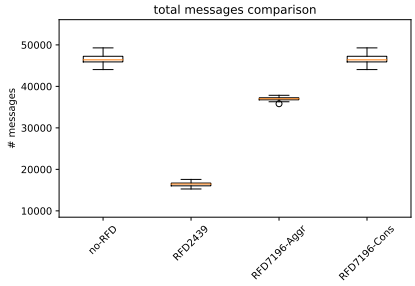
(k) Number of messages respect to the RFD strategy, MRAI=30s



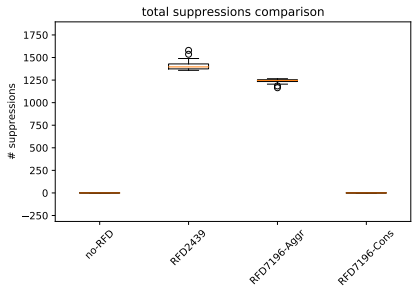
(l) Number of suppressions respect to the RFD strategy, MRAI=45s



(m) Convergence time respect to the RFD strategy, MRAI=60s

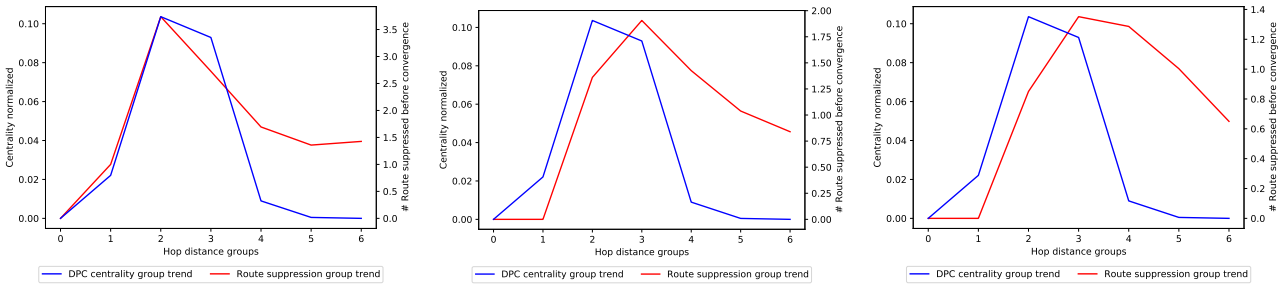


(n) Number of messages respect to the RFD strategy, MRAI=60s



(o) Number of suppressions respect to the RFD strategy, MRAI=60s

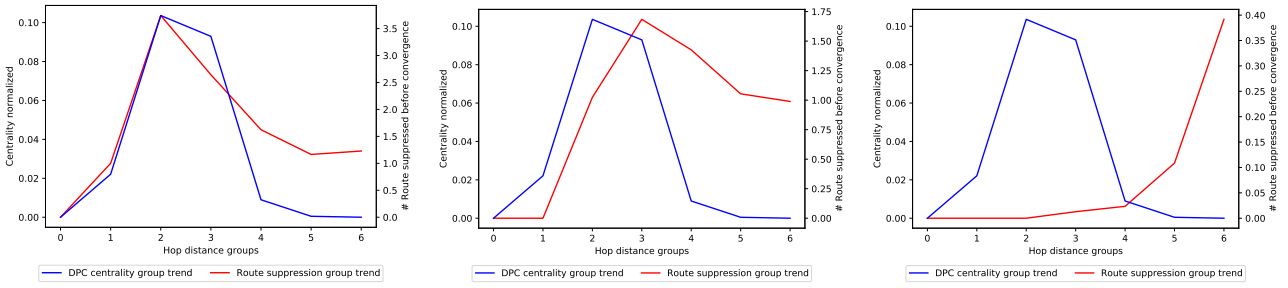
Figure A.8: Internet like topology 1000 nodes, random destination, 5 flaps, 300s delay, Network performances



(a) RFD 2439 Strategy, MRAI=0s

(b) RFD 7196 Aggressive Strategy, MRAI=0s

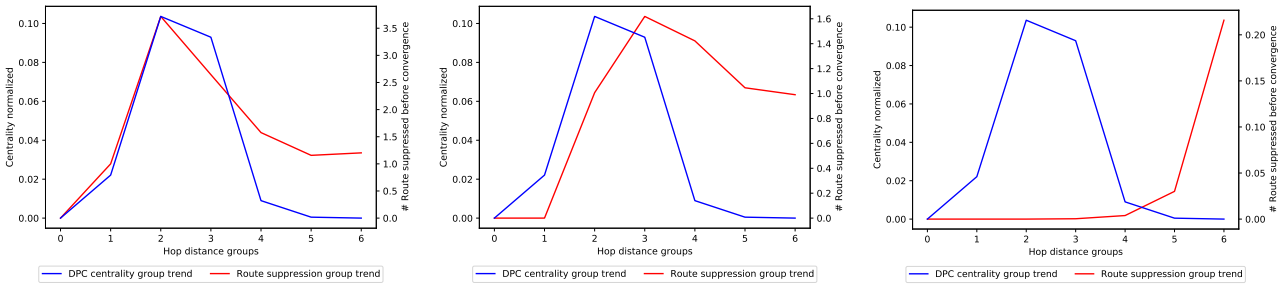
(c) RFD 7196 Conservative Strategy, MRAI=0s



(d) RFD 2439 Strategy, MRAI=15s

(e) RFD 7196 Aggressive Strategy, MRAI=15s

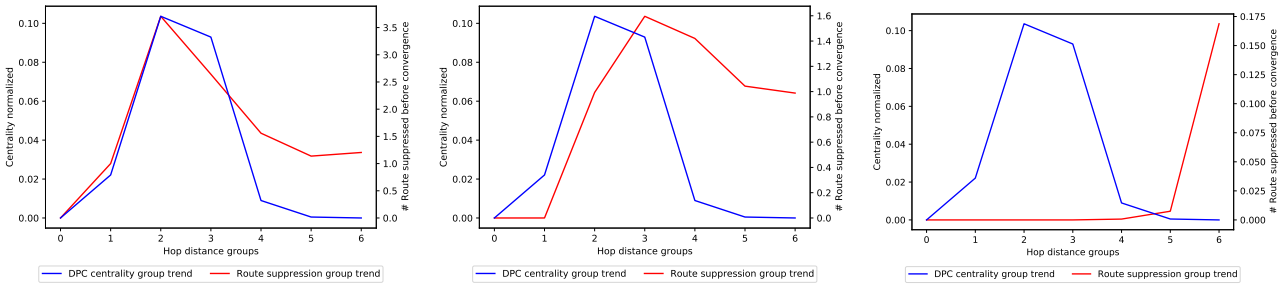
(f) RFD 7196 Conservative Strategy, MRAI=15s



(g) RFD 2439 Strategy, MRAI=30s

(h) RFD 7196 Aggressive Strategy, MRAI=30s

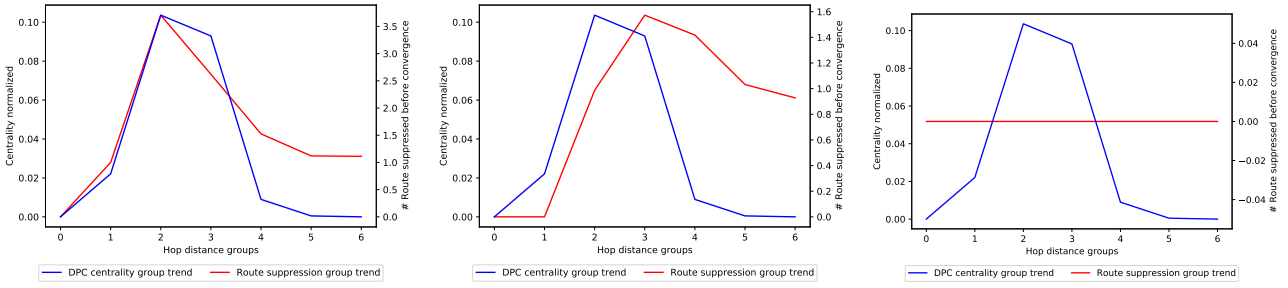
(i) RFD 7196 Conservative Strategy, MRAI=30s



(j) RFD 2439 Strategy, MRAI=45s

(k) RFD 7196 Aggressive Strategy, MRAI=45s

(l) RFD 7196 Conservative Strategy, MRAI=45s

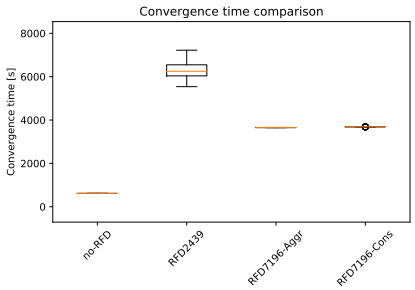


(m) RFD 2439 Strategy, MRAI=60s

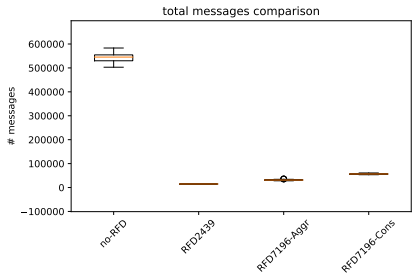
(n) RFD 7196 Aggressive Strategy, MRAI=60s

(o) RFD 7196 Conservative Strategy, MRAI=60s

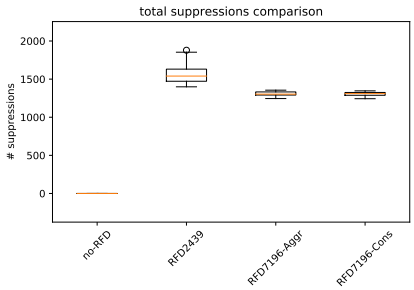
Figure A.9: Internet like topology 1000 nodes, random destination, 5 flaps, 300s delay, Suppression trend VS avg hop centrality



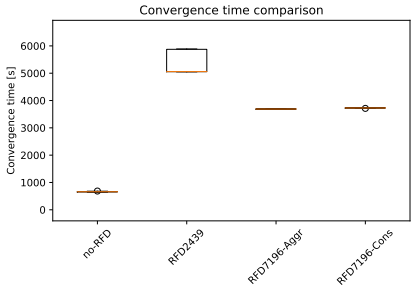
(a) Convergence time respect to the RFD strategy, MRAI=0s



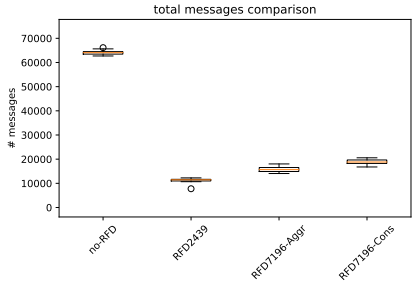
(b) Number of messages respect to the RFD strategy, MRAI=0s



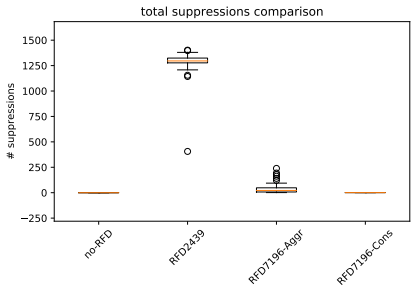
(c) Number of suppressions respect to the RFD strategy, MRAI=0s



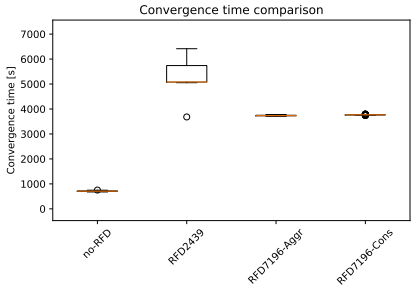
(d) Convergence time respect to the RFD strategy, MRAI=15s



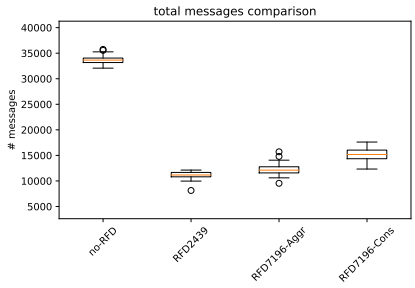
(e) Number of messages respect to the RFD strategy, MRAI=15s



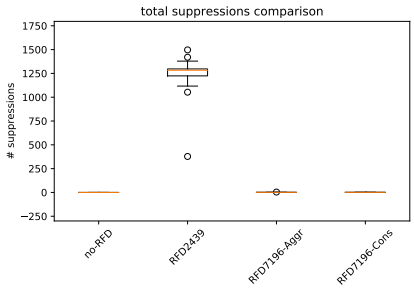
(f) Number of suppressions respect to the RFD strategy, MRAI=15s



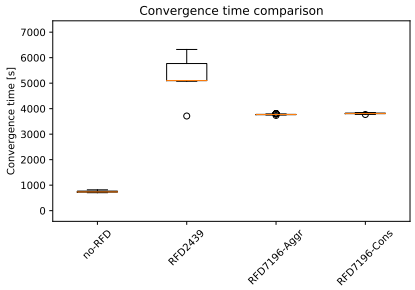
(g) Convergence time respect to the RFD strategy, MRAI=30s



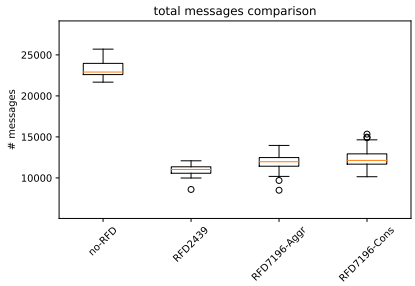
(h) Number of messages respect to the RFD strategy, MRAI=30s



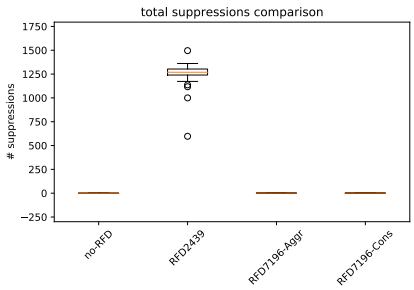
(i) Number of suppressions respect to the RFD strategy, MRAI=30s



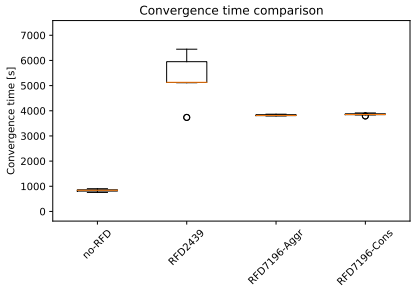
(j) Convergence time respect to the RFD strategy, MRAI=45s



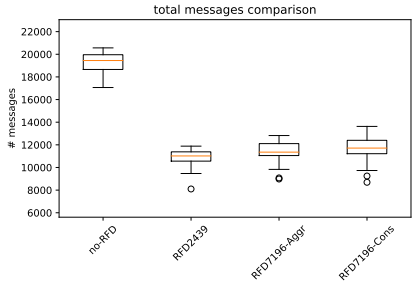
(k) Number of messages respect to the RFD strategy, MRAI=30s



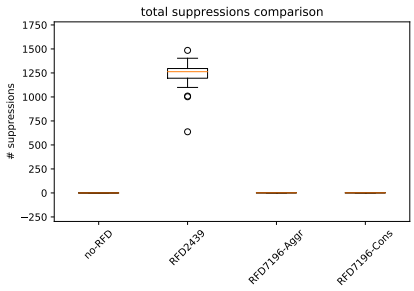
(l) Number of suppressions respect to the RFD strategy, MRAI=45s



(m) Convergence time respect to the RFD strategy, MRAI=60s

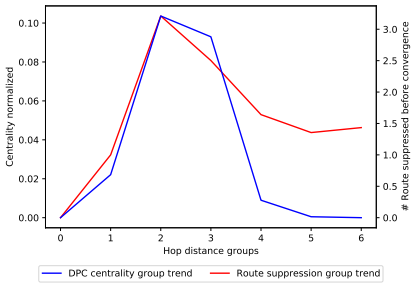


(n) Number of messages respect to the RFD strategy, MRAI=60s

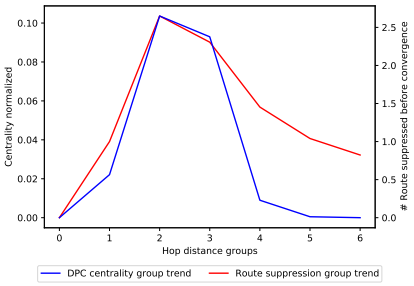


(o) Number of suppressions respect to the RFD strategy, MRAI=60s

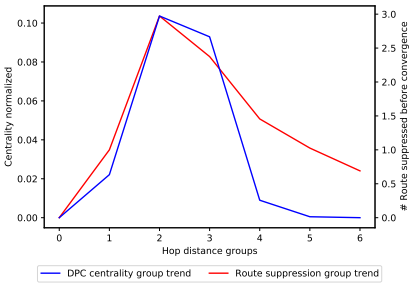
Figure A.10: Internet like topology 1000 nodes, random destination, 100 flaps, 3s delay, Network performances



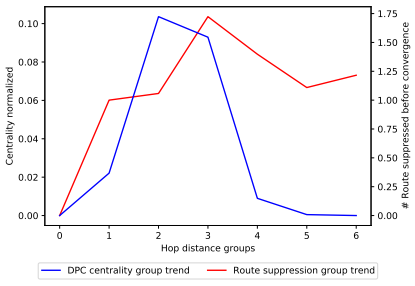
(a) RFD 2439 Strategy, MRAI=0s



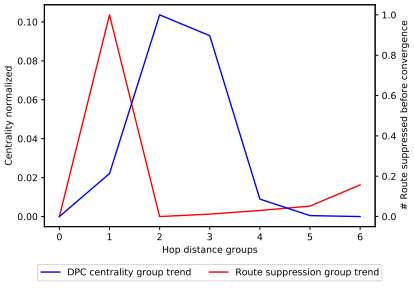
(b) RFD 7196 Aggressive Strategy, MRAI=0s



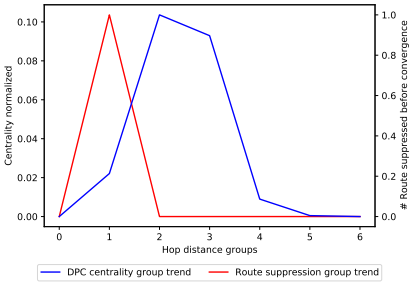
(c) RFD 7196 Conservative Strategy, MRAI=0s



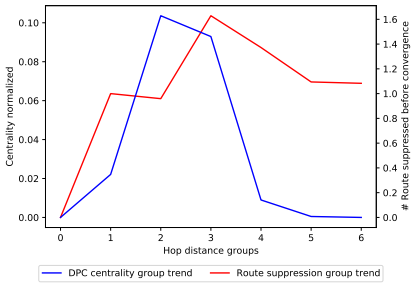
(d) RFD 2439 Strategy, MRAI=15s



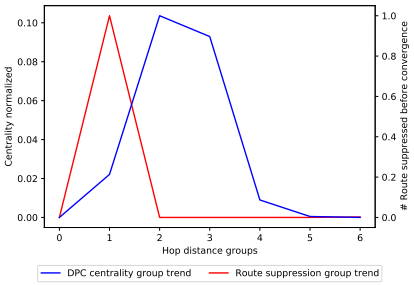
(e) RFD 7196 Aggressive Strategy, MRAI=15s



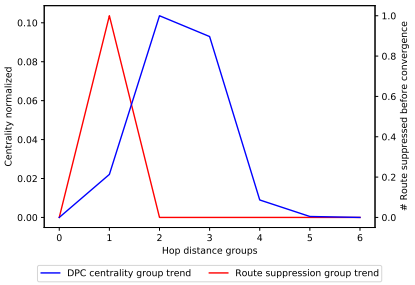
(f) RFD 7196 Conservative Strategy, MRAI=15s



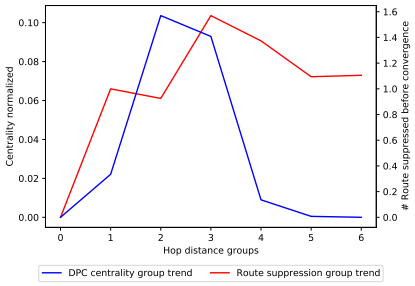
(g) RFD 2439 Strategy, MRAI=30s



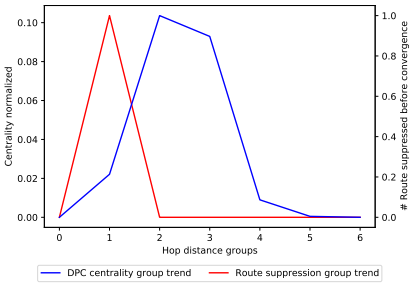
(h) RFD 7196 Aggressive Strategy, MRAI=30s



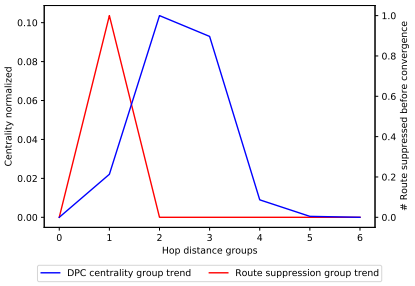
(i) RFD 7196 Conservative Strategy, MRAI=30s



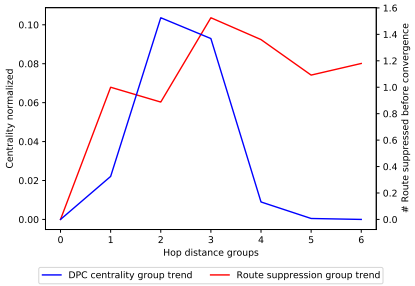
(j) RFD 2439 Strategy, MRAI=45s



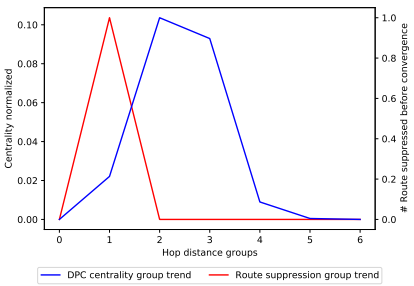
(k) RFD 7196 Aggressive Strategy, MRAI=45s



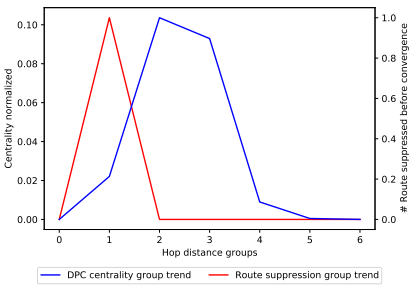
(l) RFD 7196 Conservative Strategy, MRAI=45s



(m) RFD 2439 Strategy, MRAI=60s



(n) RFD 7196 Aggressive Strategy, MRAI=60s



(o) RFD 7196 Conservative Strategy, MRAI=60s

Figure A.11: Internet like topology 1000 nodes, random destination, 100 flaps, 3s delay, Suppression trend VS avg hop centrality

Abbreviations

ADV advertisement

AS Autonomous System

BGP Border Gateway Protocol

c2p customer-to-provider

CAIDA Center for Applied Internet Data Analysis

CFSM Communicating Finite-State Machine

DES Discrete Event Simulator

DPC Destination Partial Centrality

DV Distance Vector

eBGP Exterior BGP

FSM Finite State Machine

iBGP Interior BGP

ISP Internet Service Providers

IW Implicit Withdraw

LS Link State

MRAI Minimum Route Advertisement Interval

p2p peer-to-peer

PV Path vector

RFC Request For Comment

RFD Route Flap Damping

RIB Routing Information Base

RNG Random Number Generator

s2s sibling-to-sibling

SPP Stable Paths Problem

SSP Stratified Shortest Path

TCP Transmission Control Protocol