UNIVERSITÀ
DI TRENTO

Dept. of Information Engineering and Computer Science

Master's Degree in
Computer Science

FINAL DISSERTATION

TITLE

Subtitle (optionl)

Supervisors
……

graduating student
Milani Mattia

Accademic Year 2019/2020

Ringraziamenti

...thanks to...

Contents

Summary

...summary....

Minimum Route Advertisement Interval (MRAI)

1 Introduction

- How is internet built
- the protocol that controls internet

1.1 Internet nowadays

- Use today studies to show how internet is today

1.2 Correlation between variables and convergence

- Expose the hypothesis of the correlation

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

- Why is important understand this correlation?

2 BGP state of the art

- BGP de facto standard on the internet
- What is an AS
- interconnection between ASes

2.1 BGP

- High level of BGP
- BGP messages
- BGP Update messages
- BGP policies

2.2 BGP Wedgies

- What are wedgies?
- why are them important?
- which situations them occur?

2.3 BGP MRAI

- What is MRAI?
- Previous works on MRAI
- Suppositions on the MRAI influence

2.4 BGP RFD

- What is RFD?
- Why is used RFD?
- Evolution of RFD?
- RFD Today

3 Discrete Event Simulator

Experiments on Border Gateway Protocol (BGP) are not applicable on the Internet, for this reason different studies shows their results using a simulate environment [1] 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 Autonomous System (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 [2, 3].

I developed the DES as a highly modular environment. 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 descriptor of the network. it defines also all topological information and all the properties of each 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 [4].

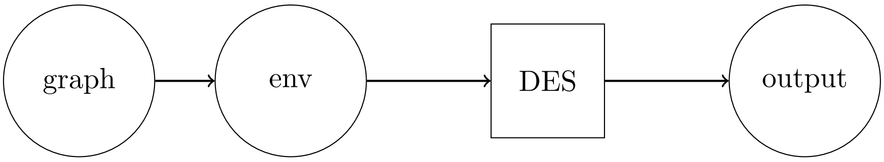


Figure 3.1: Discrete event simulator structure

```
<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 are 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 contains a destination, for example, the signal "AWA" defines that there will be an announce followed by a withdraw and the an other 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 contains 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 combination 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.

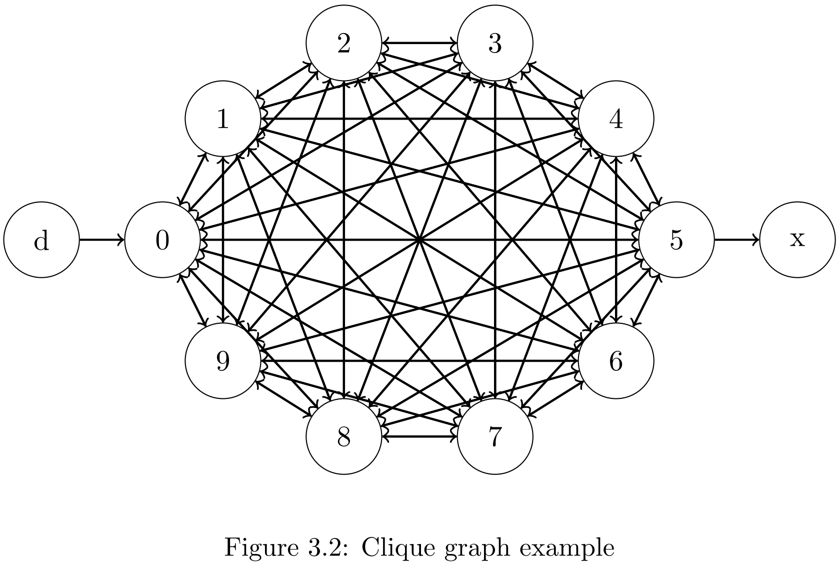


Figure 3.2: Clique graph example

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 [5]. 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 choose of MRAI is fundamental for the network convergence. An example of the network used is presented in Figure 3.3.

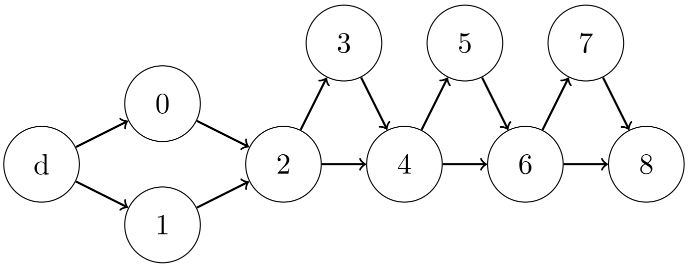
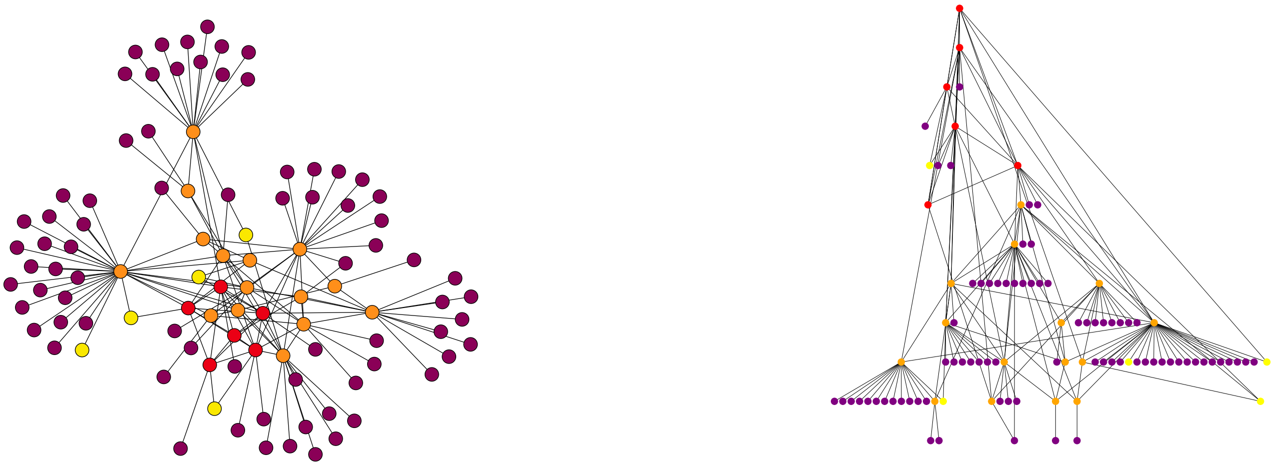


Figure 3.3: Fabrikant chain graph example

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 then 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. [6] 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 Protocols 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 to reach a single destination. Transitions encode the messages that a node has received to change state, on the edges are also inserted the response messages that the node will transmit.

FiXme: Insert image and table for an example

In BGP messages transmit 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 we can see multiple messages, for example "A1W1A1", that will be compressed in "A1" and then evaluated.

The concept for a BGP FSM has been "taken" from [7].

- BGP as an FSM main idea
- signaling transmutation

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 showed in [7]. The graph used for the study is presented in Fig. 4.1.

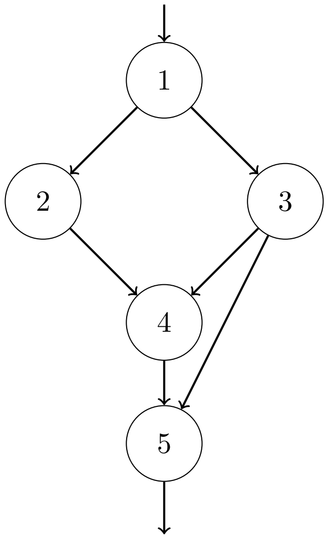


Figure 4.1: Graph from fig 4 of [7] used to study the FSM of the nodes

This topology, Figure 4.1, present an Stable Paths Problem (SPP) with five nodes [8]. 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 an hypothetical destination and it will spread this information through an advertisement (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.

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

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

FiXme: this paragraph is cumbersome The two nodes of more interest are node 4 and node 4. The first one can receive multiple combination 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.

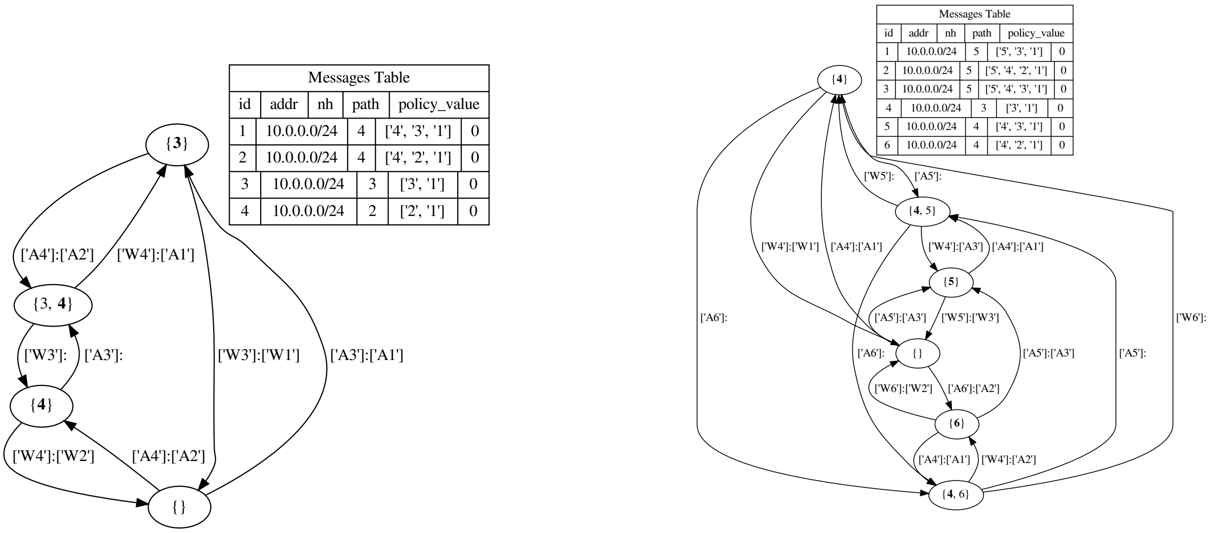
From the 100 total runs we can generate the CFSM of node 4 and node 5, in order to be able to study how the nodes reacts to different input signals. The two CFSM are presented in Figure 4.2.

FiXme: Remove message table from Figure 4.2?

The states of the CFSM in Figure 4.2 are represented by the knowledge of the nodes, composed by the routes that are in the Routing Information Base (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 it’s neighbours, for an example take a look to Figure 4.2a 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 see in Figure 4.2b the transition from $\{1, 4\}$ to $\{1, 3\}$ thanks to the reception of the announcement $a3$ from the node 4.

As written in [7], 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 provoke all the transitions.



Messages Table				
id	addr	nh	path	policy_value
1	10.0.0.0/24	4	['4', '3', '1']	0
2	10.0.0.0/24	4	['4', '2', '1']	0
3	10.0.0.0/24	3	['3', '1']	0
4	10.0.0.0/24	2	['2', '1']	0

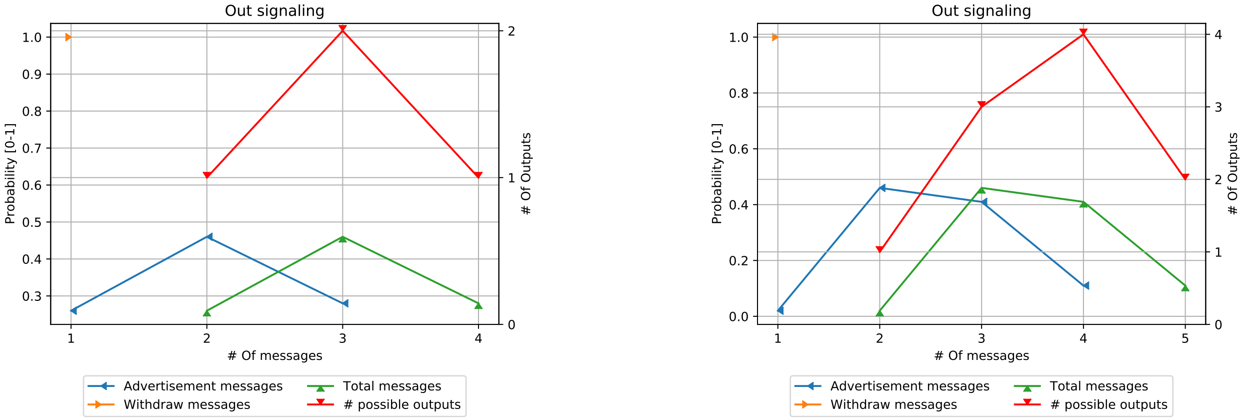
Messages Table				
id	addr	nh	path	policy_value
1	10.0.0.0/24	5	['5', '3', '1']	0
2	10.0.0.0/24	5	['5', '4', '2', '1']	0
3	10.0.0.0/24	5	['5', '4', '3', '1']	0
4	10.0.0.0/24	3	['3', '1']	0
5	10.0.0.0/24	4	['4', '3', '1']	0
6	10.0.0.0/24	4	['4', '2', '1']	0

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

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

Figure 4.2: CFSM of nodes 4 and 5 of the graph Figure 4.1 with an input signal of "AW"

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 4.3: Output signal study of nodes 4 and 5 of the graph Figure 4.1 with an input signal of "AW" at node 1

The plots in Figure 4.3 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 represent the number of messages in the output signal, a message is a single announcement or withdraw. The first y axis represent 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 refers to it, the blue one represent the number of advertisement messages in the output signal correlated with the respective probability. For example in Figure 4.3a 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 represent 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 4.3b 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 does 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

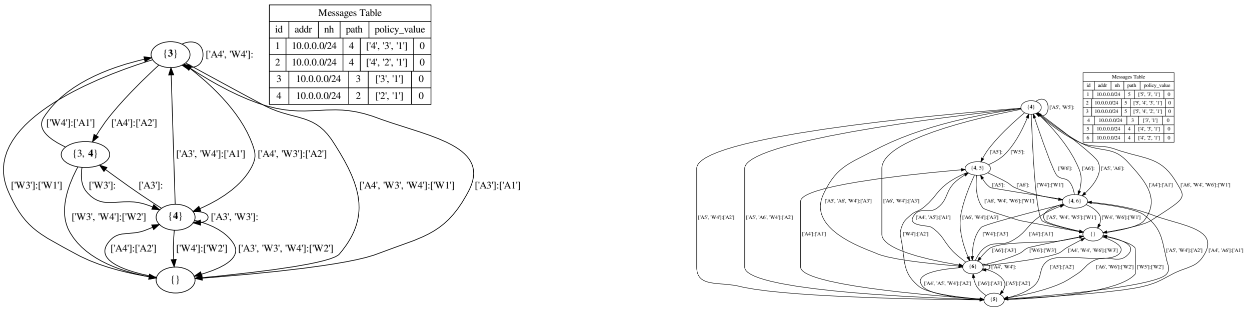
Table 4.3: Node 4 and 5 different output signals encountered during the runs

4.2.1 MRAI and BGP FSM

How would MRAI affect the study of the signals produced by Figure 4.1? 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 4.4.

FiXme: Figure 4.4b is not readable at all, move the two figure one after the other



Messages Table				
id	addr	nh	path	policy_value
1	10.0.0.0/24	4	['4', '3', '1']	0
2	10.0.0.0/24	4	['4', '2', '1']	0
3	10.0.0.0/24	3	['3', '1']	0
4	10.0.0.0/24	2	['2', '1']	0

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

Messages Table				
id	addr	nh	path	policy_value
1	10.0.0.0/24	5	['5', '3', '1']	0
2	10.0.0.0/24	5	['5', '4', '3', '1']	0
3	10.0.0.0/24	5	['5', '4', '2', '1']	0
4	10.0.0.0/24	3	['3', '1']	0
5	10.0.0.0/24	4	['4', '3', '1']	0
6	10.0.0.0/24	4	['4', '2', '1']	0

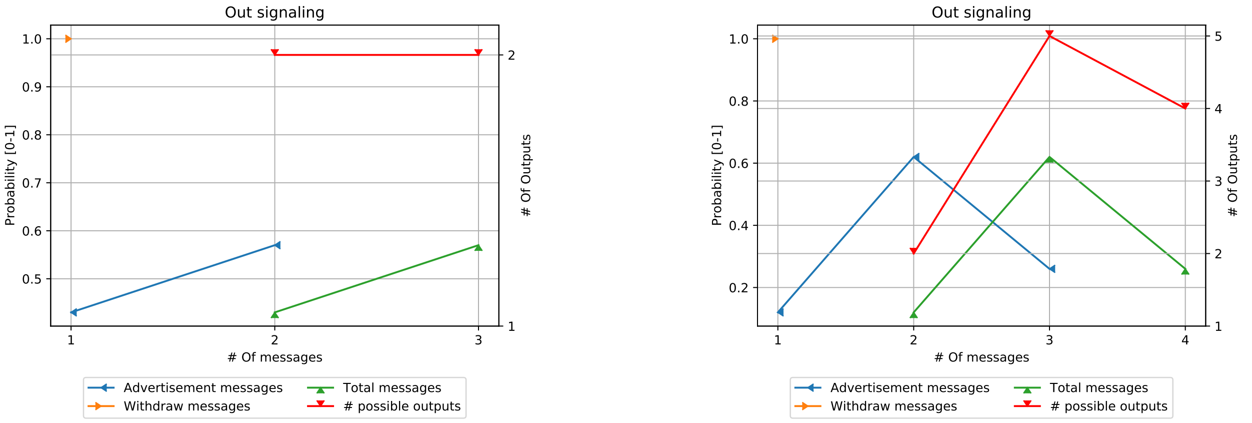
(b) Node 5 CFSM from the environment of Table 4.1 with MRAI=30 s

Figure 4.4: CFSM of nodes 4 and 5 of the graph Figure 4.1 with an input signal of “AW” with MRAI=30 s

Figure 4.2a and Figure 4.4a 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 4.5.

Comparing Figures 4.3b and 4.5b 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.



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

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

Figure 4.5: Output signal study of nodes 4 and 5 of the graph Figure 4.1 with an input signal of "AW" at node 1 with MRAI=30 s for every link

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. [5] 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 accept and publish 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 [9].

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 resemble the study conducted by [5] 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 Implicit Withdraw (IW) capability of BGP. This parameter will influence the number of messages that will be transmitted.

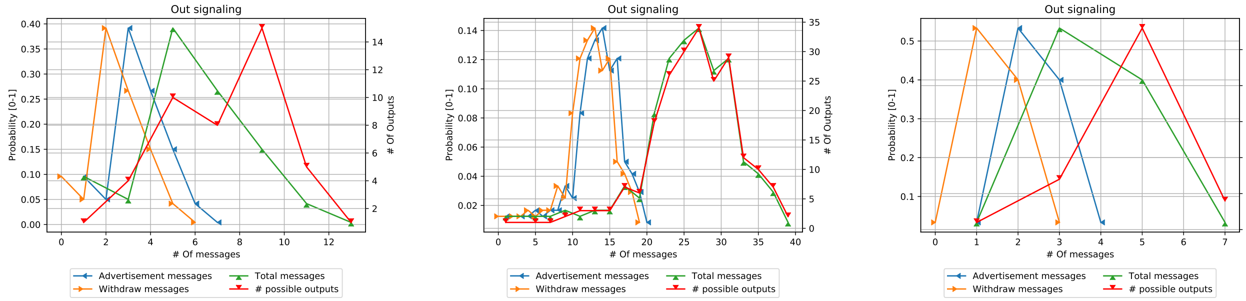
The results of all those different experiments, in terms of CFSM are exposed in Table 4.5

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 gray 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 [5] 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 combination of input signals for the nodes. The opposite case in respect of the *Descendent* strategy obtain 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 take 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 4.6.



(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 4.6: 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 4.6a is the one that represent the actual standard of the protocol [10]. 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. 4.6b where the maximum length of output signal reaches almost 40 messages, but the most probable output signal have 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 4.6c 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 [5], 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 dependance

MRAI is one of the parameters that mostly has caused divergences in the scientific community. And, after the introduction in the protocol since the version 4 [10] 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 [5].

The protocol strictly depends on this parameter, because as we saw in Chapter 4, the incorrect use of it can lead to tremenodous consequences, even worst of not having it at all. In other cases, with a particular setting of it is possible to improve the network perforcances. Recent studies about centrality metrics on routing protocols introduce, through the distributed computation of the metric, to a timer trade off improvement [11, 12]. This kind of approach has been also applied on BGP with positive results on network failures [13, 14].

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 scenario as specified in Labovitz et al. [15] I used two approaches in this Environment, the first one keeps the IW active the second one avoid the 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

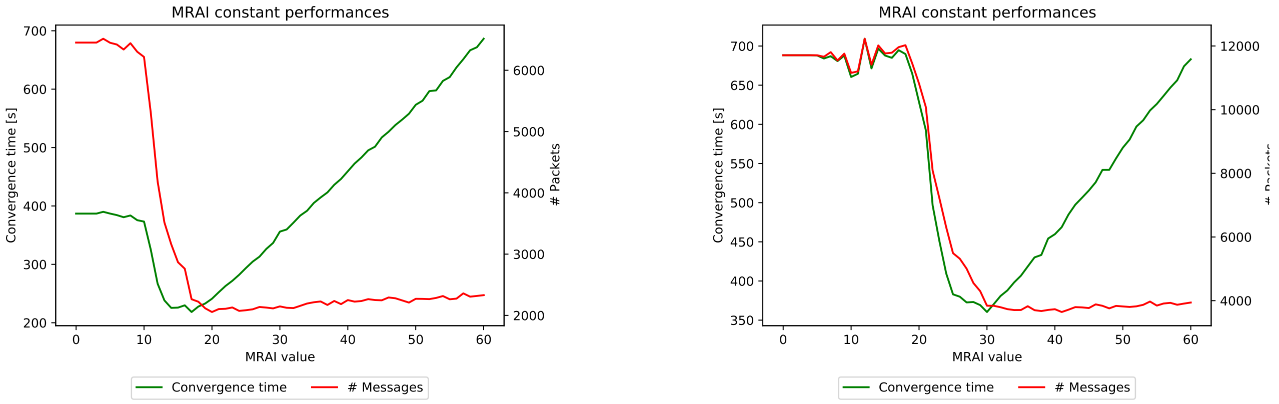
Property	Value
Seeds	$[1, 10]$
Signaling	"AW"
Withdraws delay	Uniform distribution between 1 s and 5 s
Announcement delay	constant distribution of 5 s
MRAI	$[0, 60]$
Link delay	Uniform distribution between 0.0001 s and 0.5 s

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 represent the number of messages transmitted, the red line.

The effets 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 perforcances **with IW**

(b) Network perforcances **withouth 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 starts to wait more time withouth obatining more useful information. This can be see 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 ressembling the Elmokashfi properties [6] already described in Section 3.1.3.

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

5.3 Strategy dependence

Like I mentioned before, the network perfomances depends on the MRAI strategies choosen. For this reason the first point of my study is to point out this differences. In order to do that, the first study that I would like to present is the one that study how the standard protcol 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 betwen 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

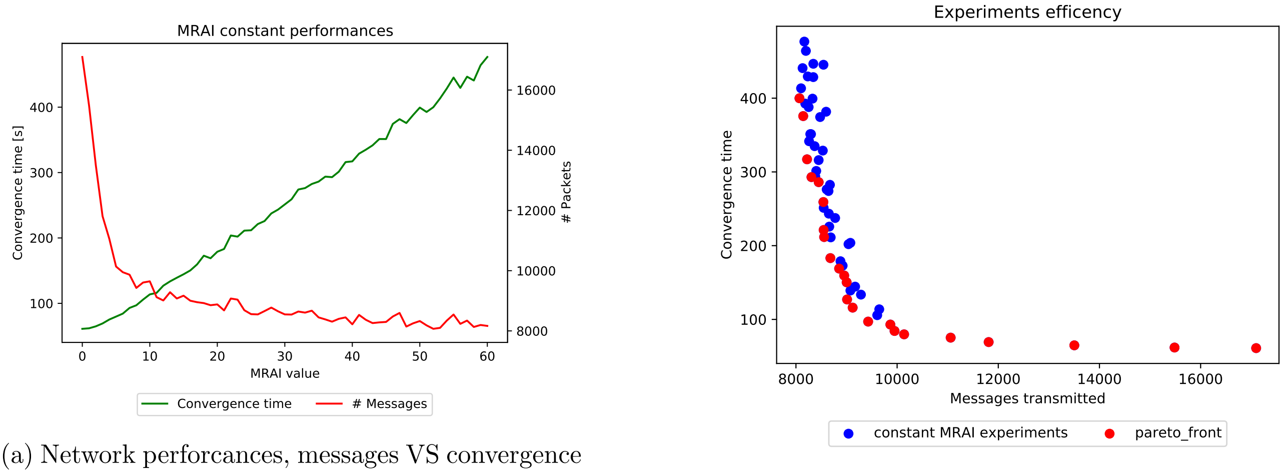
As MRAI strategies I decided to use the following two:

- ***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 [14]

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 [13] I will assume that is precalculated and that every node knows it's own centrality to set the timers.

To permit a comparison between thise 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 intrisically 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 perforcances, messages VS convergence time with different MRAI values

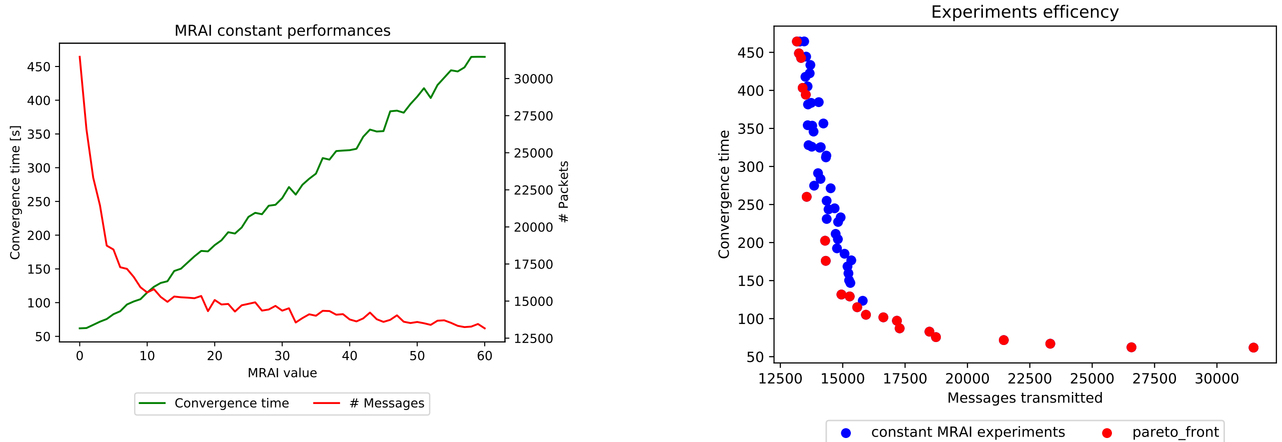
(b) Pareto front of Messages VS Convergence time

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 ammount 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 happend for the clique graph in Figure 5.1. This continious linear grow is dictated by the fact that nodes keep meaningfull 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 efficent, this concept has been already used in engineering to define the set of best outcomes from the tradeoff of two different parameters [16]. We can clearly see that the majority of the points is concentrated to the left of the chart, this means that few MRAI values would give as 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 withouth 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.

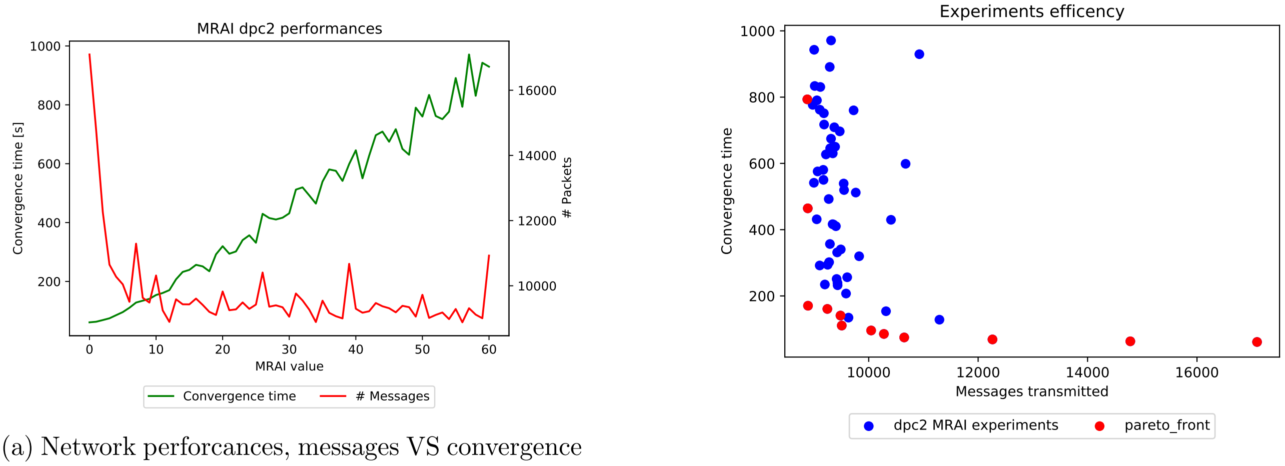


(a) Network perforcances, messages VS convergence time with different MRAI values

(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. **Withouth 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 valure are comparable one another.



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

(b) Pareto front of Messages VS Convergence time

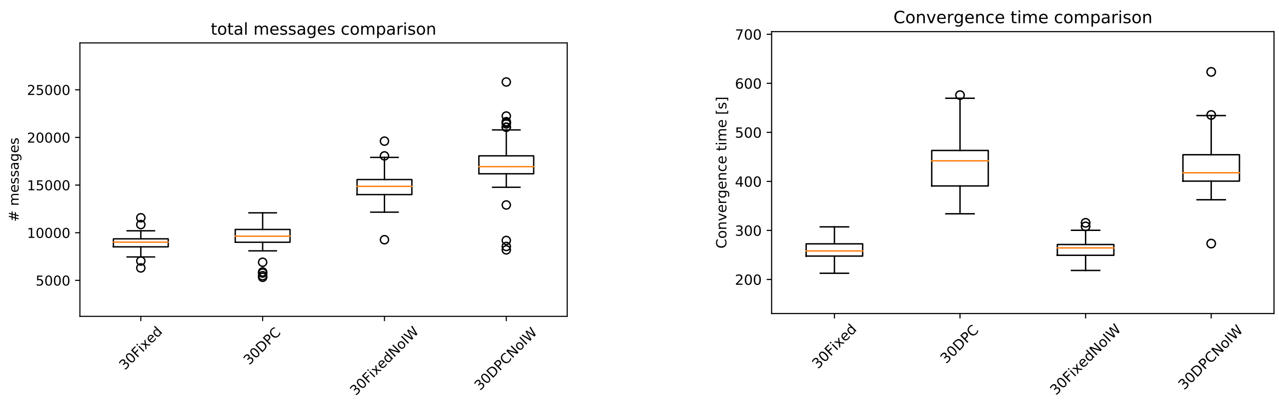
Figure 5.4: Evolution of the network performances on the **Internet Like** graph of 1000 nodes using a *DPC* MRAI strategy with an $MRAI_{mean}$ from 0 to 60 seconds.

FiXme: Combersome, 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 leads to a different trend. Both are equal at the begininngin 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 variates more and it converge 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 [10] so I compared those strategies performances in a boxplot in Figure 5.5. I decided to run 100 different runs for each strategy with the $MRAI_{mean}$ fixed to 30 s.



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

In Figure 5.5 we can compare those two strategies, the first figure, Figure 5.5a represent the number of messages trasmitted by the 100 runs, we can see that the two strategies, withouth IW, are really close one another. While in the time required for convergence, Figure 5.5b there are some huge difference between the two strategies, is not negligeable that with the DPC strategy the time required is almost the double of the standard time.

In cocnlusion we can say that the MRAI strategy is one of the factor that can influence the Network perfomances.

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 possiblities that are generated through the pareto efficiency plot and compare the results with the pareto efficency graphs. To permit this comparison I would set MRAI randomly but like for the DPC strategy respecting the average required.

FiXme: Insert table with the value of MRAI used and the ranges, and the number of total experiments executed

FiXme: Insert the resulting plot

5.5 Signal dependance

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

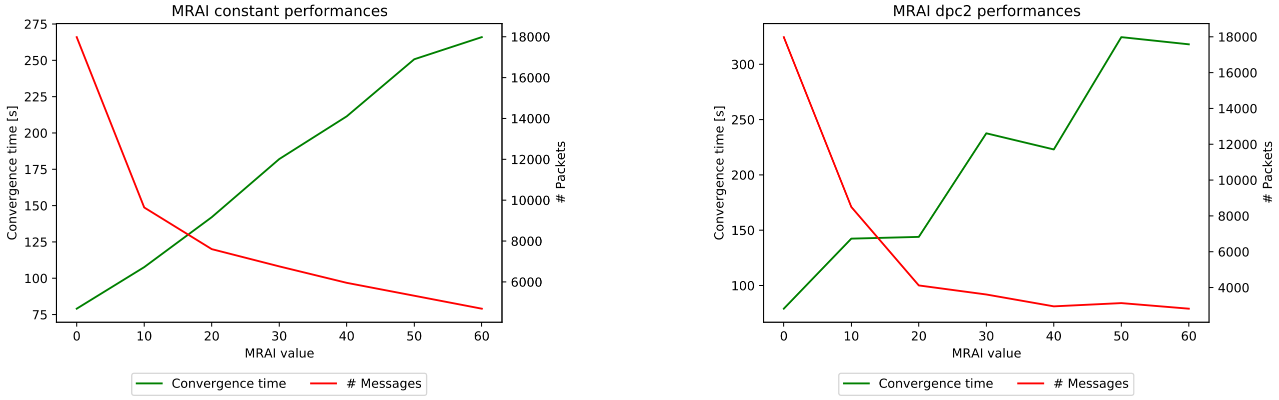
For this reason I used the same environment described before and execute the experiments with different input signals from the same node, “AWA”, “AWAW” and “AWAWA”.

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

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

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

In Figure 5.6 is possible to see the evolution for the signal “AWA”.



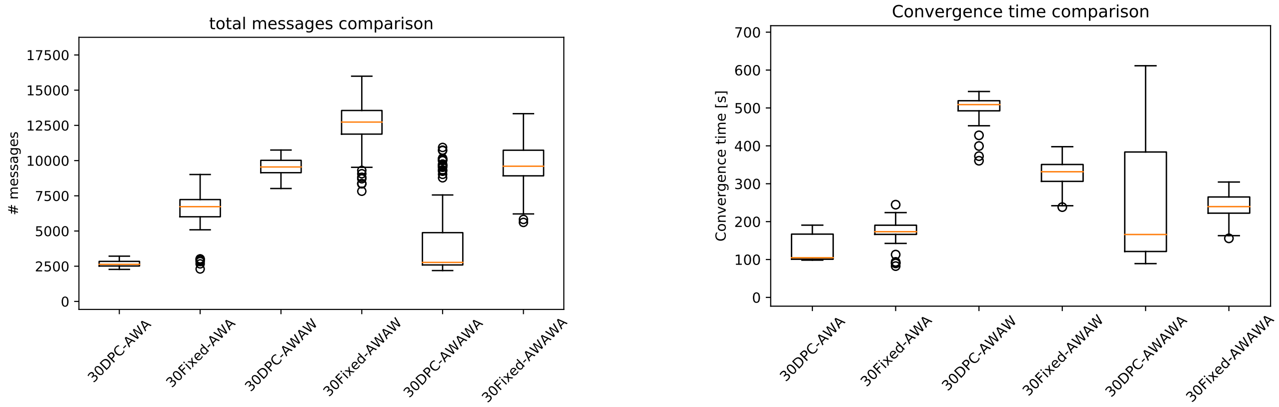
(a) Network perforcances, *fixed* MRAI strategy

(b) Network perforcances, DPC MRAI strategy

Figure 5.6: Network perfomances comparison with different MRAI strategies, Graph internet like with 1000 nodes, signal "AWA"

Is possible to notice in Figure 5.6 a huge difference in respect of the plots in Figures 5.2 and 5.4. The DPC strategy was abele to outcome the standard *fixed* strategy over multiple prospective. Analyzing Figure 5.6b 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 MRAI timer *mean* of 30 s the number of messages is less than 1/4 in respect of an MRAI mean of 0 s. The convergence time courve has a compleatly 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. Infact some nodes are able to cache both the "WA" part of the signal and compleatly avoid to send anything at all, because them have already transmitted the first "A". The compleate compression of the signal "AWA" is "A". The other evolutions, 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 differnet runs for each strategy and each different signal, the results are exposed in Figure 5.7b.



(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 5.7: 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 "allSignals"

Is possible to notice in Figure 5.7 that both the strategies has 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 an other "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 an other message to notify the withdraw. And withdraws are not affected by MRAI like specified in a IETF draft of 2012 named

“Revisions to the BGP ’Minimum Route Advertisement Interval”’.

Other than that is possible to notice that the DPC techniques has better restults in terms of messages transmitted, while it could have an 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

5.6 Position dependance

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 ipothesis 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 with different nodes that provoke messages storms.

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 its possible to define different levels of the graph. If we take the central clique as the rout 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 on the same way?

- And how much is influent the position?
- Hierarchically?

6 RFD and MRAI correlation

Route Flap Damping (RFD) is another parametr of BGP used to avoid messages storms. It is used to avoid flapping routes to continously 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 (ora after a certian time).

RFD, other than MRAI, is one of the most studied parameters of BGP because of its influence in the convergence time [17,18]. RFD received different updates from its first implementation, but recent studies showed that most of the providers still use outdated parameters [19].

The use of deprecated values can lead to a to havy restrictive suppression of small routes delaying the correct spreading of information. Some cases of suppression are caused by faulty interfaces that haviliy 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 flapss and how would the new versione of RFD reacted to it. Finally, I would look forward to understand the correlation between RFD and MRAI. When a suppressed route is shared again it could provoke messages storm that triggers different MRAI session, or the opposite case, a low MRAI that cause the grow of the figure of merit that suppress a route.

6.1 RFD on toy topologies

- What is the impact of RFD?
- In which occasion is present RFD?
- Clique
- Variations thanks to MRAI

6.2 RFC 2439 VS RFC 7196

- Time comparison between both of them
- how them react differently?
- why?

6.3 Mice VS Elephants

- What is Mice VS Elephants?
- How has been studied in the past?
- Introduce how MRAI affects mice VS elephants

7 Conclusion

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

7.1 Future Works

:)

Bibliography

[1] 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.

[2] 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.

[3] 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.

[4] 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.

[5] 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.

[6] 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.

[7] T. G. Griffin, “A Finite State Model Update Propagation for Hard-State Path-Vector Protocols.”

[8] 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.

[9] 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.

[10] 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.

[11] 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.

[12] 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.

[13] 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

[14] 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.

[15] 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.

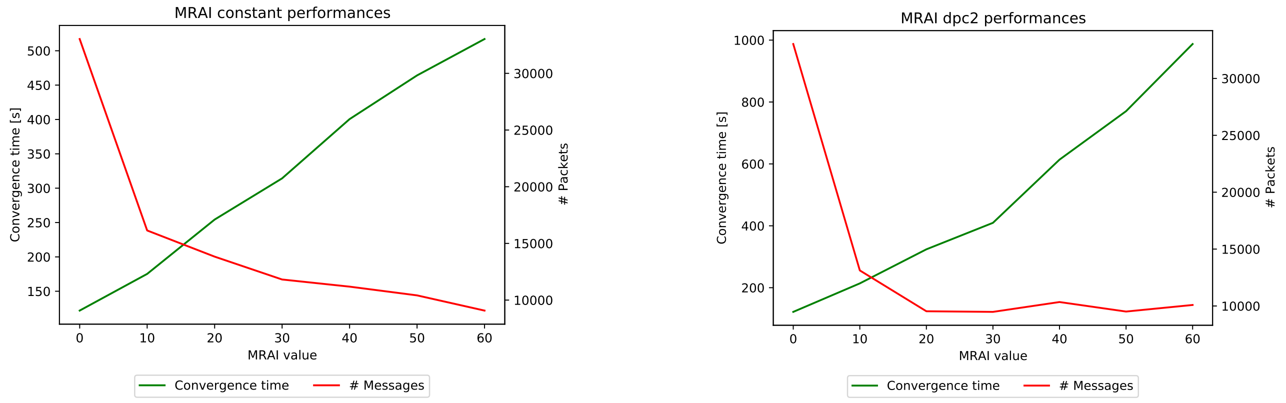
[16] E. Goodarzi, M. Ziaei, and E. Z. Hosseinipour, *Introduction to optimization analysis in hydrosystem Engineering*. Springer, 2014.

[17] 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.

[18] 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.

[19] 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.

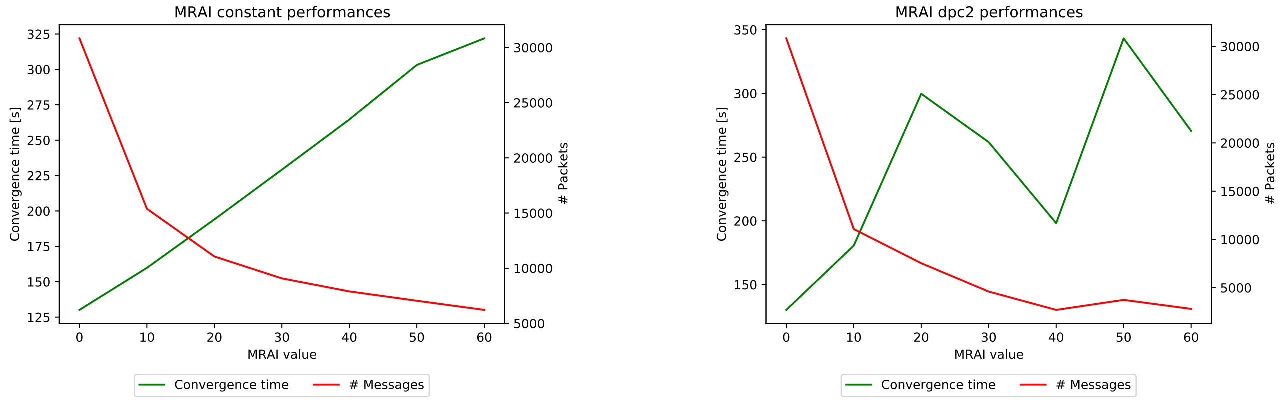
Appendix A Appendix



(a) Network perforcances, *fixed* MRAI strategy

(b) Network perforcances, DPC MRAI strategy

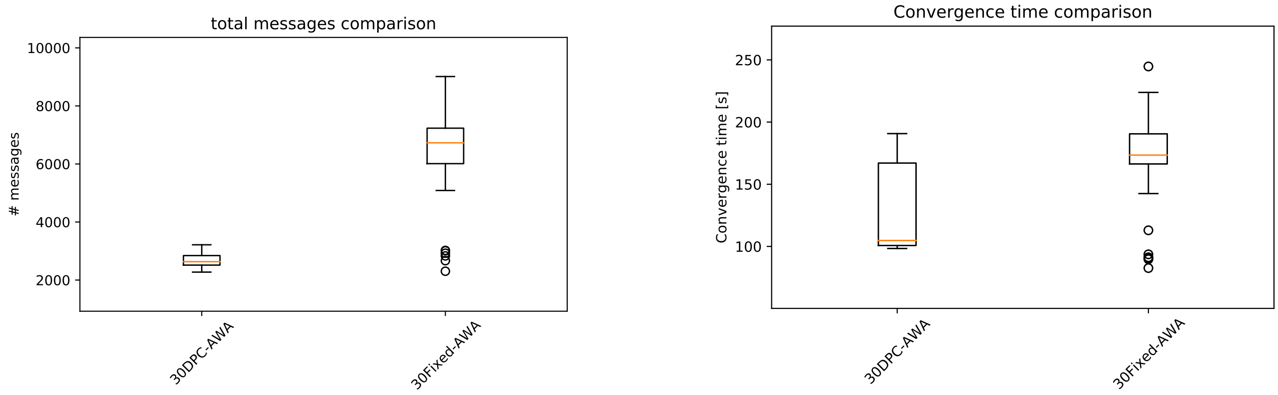
Figure A.1: Network perfomances comparison with different MRAI strategies, Graph internet like with 1000 nodes, signal “AWAW”



(a) Network perforcances, *fixed* MRAI strategy

(b) Network perforcances, DPC MRAI strategy

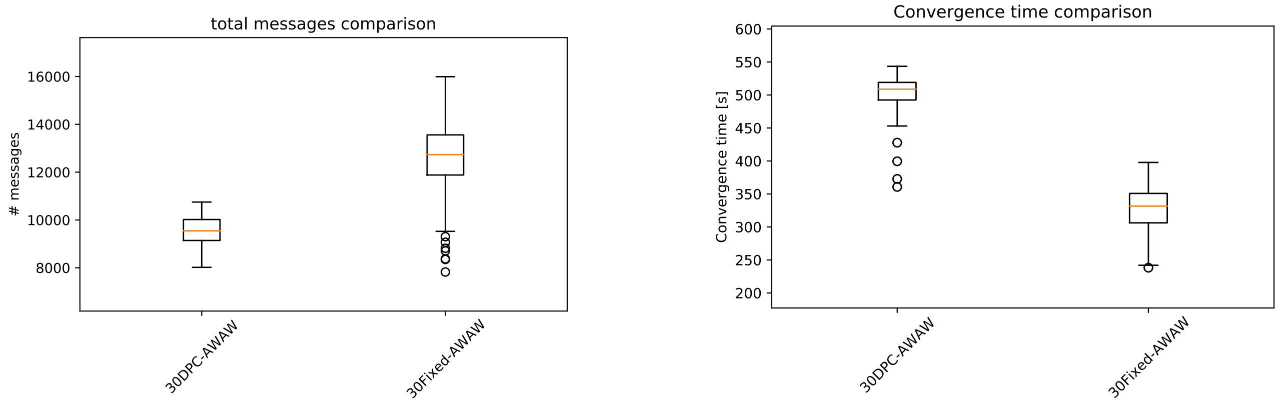
Figure A.2: Network perfomances comparison with different MRAI strategies, Graph internet like with 1000 nodes, signal “AWAWA”



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

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

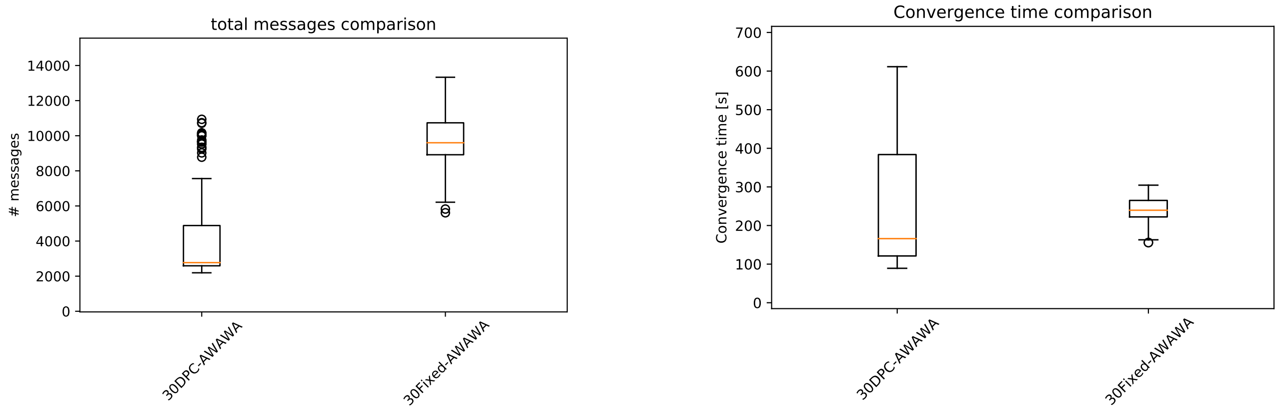
Figure A.3: Network perfomances comparison with different MRAI strategies, Graph internet like with 1000 nodes, MRAI value 30 s, number of runs for each strategy 100, signal "AWA"



(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"

Abbreviations

ADV advertisement

AS Autonomous System

BGP Border Gateway Protocol

CFSM Communicating Finite-State Machine

DES Discrete Event Simulator

DPC Destination Partial Centrality

FSM Finite State Machine

IW Implicit Withdraw

MRAI Minimum Route Advertisement Interval

RFD Route Flap Damping

RIB Routing Information Base

RNG Random Number Generator

SPP Stable Paths Problem