SumUp 14-10-2020

Mattia Milani*
*Dept. of Information Engineering and Computer Science, University of Trento, Italy
mattia.milani@studenti.unitn.it

I. Experiments Presentation

The goal of this document is, to sum up, and describe the experiments done up to now. All the experiments were done using the software in this repository and are fully replicable. How to run and analyze the experiments is out of the scope of this document.

The experiments are divided in two main categories:

- ***Single node evaluation***, in this group of experiments the goal is to analyze a single node evolution in the network;
- ***Network evaluation***, in this group of experiments the evaluation is done on the entirety of the network.

II. Goals

Like I specified in Section I all the experiments are divided into two categories that are distinguished by the size of the analysis. The simulation environment could be the same but the difference is in the analysis of the evolution.

In the first case, ***Single node evaluation***, the goal of the analyzer is to study a specific node and highlight the evolution of it. The output of the analysis could be the Finite State Machine (FSM) of the node and the signalling plot.

The signalling of a node represents all the possible outputs of a node. A single output signal represents in a single experiment the messages transmitted by the node, the result will be a mix of advertisements and withdraws in a string like "A1W1A4A6W6" This ouptuts signals, for each bunch of experiments, are collected in a CSV file with the appearance frequency of each output signal.

In the second case, ***Network evaluation***, we are looking for network results, evaluating the entire set of nodes and links. This is done by studying the number of messages transmitted and the convergence time.

Given T_{tx} as the time of the first transmission and T_{rx} as the time of the **last** reception the convergence time, CT, is given by the delta of those times: $CT = T_{rx} - T_{tx}$. The convergence is reached when the network becomes silent again.

III. Environments

Multiple environments have been used for the experiments. The main differences and properties of those environments are described in this section.

The first environment that I used is a *Fabrikant* environment with different Minimum Route Advertisement Interval (MRAI) settings. This name comes from the particular graph used, described in Section IV. The four types of MRAI used are:

- ***Fixed 30s***, MRAI is fixed for each link to 30 seconds;
- ***No MRAI***, MRAI is fixed for each link to 0.0 seconds;

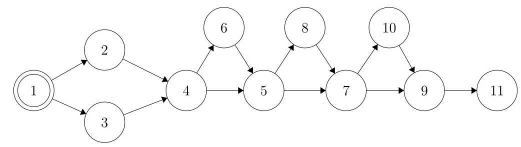


Figure 1: Fabrikant graph representation

- ***Ascendent***, MRAI will be doubled at each leach ($1-2-4-8-...$);
- ***Descendent***, Reverse of the ascendent case, MRAI will be divided by two at each leach.

The second environment used is a *clique* one, the graph is described in Section IV, all the parameters of the environment are described in Section V In all the clique experiments MRAI has been fixed in all the links. For each value of MRAI has been done 10 different experiments and MRAI goes from 0 in the first group of experiments to 60 in the last group. For a total of 610 runs in this topology.

The last one used is an *Internet Like* environment. The graph is described in Section IV, all the parameters are described in Section V. The main goal of this environment was to emulate, roughly, a real environment. The main difference between different experiments on this environment was the type of MRAI applyed:

- ***Random MRAI***, for each link the MRAI value will be chosen with a uniform distribution between $[0.0, MRAI_{limit}]$, the network must respect a defined $MRAI_{mean}$ value
- ***Fixed MRAI***, for each link the MRAI will be equal to $MRAI_{mean}$

In this environment has been run multiple experiments for each MRAI type. For each experiment, a new graph would be computed, so in the *Random MRAI*, for the same $MRAI_{mean}$ value could exists multiple different graphs.

IV. Input Graphs

In total three different base graphs has been used to produce all the results in this document.

A. Fabrikant Graph

The Fabrikant graph replicate what is described in the first figure in [1]. for simplicity, an example is reported here in Figure 1

Node 1 represent the only source of traffic, node 4 will prefer to reach the destination through node 2, but the link is slower, triggering changes in the network as required by [1].

Taking this base frabrikant graph other 4 graphs has been deveolped, one for each MRAI strategy applyed.

B. Clique Graph

The clique graph used for the clique experiemnts is composed of 15 nodes plus one external node that is the source of a destination. Each node is connected to every other node in a mesh network. The only node that does not respect this rule is the destination source. it has only one link that is connected to the node of the mesh network number 0.

Relationships between nodes are of the servicer type, so each node has 14 clients to updated when it receives an update. This ensures that the information is shared in the entire network.

For this network has been generated one graph for each fixed MRAI used, so that at the end we had 60 different clique graphs with the correct timer value equal on each link.

C. Internet Like Graph

This network is composed of 100 nodes, it is not enough to emulate the Internet but, with enough computation time, the results should be comparable with bigger graphs. The graph has been produced following [2].

In the graph has been chosen only one node that shares a destination. The node has been chosen randomly in the set of clients nodes.

For each experiment, depending on the type of MRAI, a new graph file has been generated with different MRAIs values on the edges.

In total has been generated 6000 internet like graphs.

The base graph is represented in Figure 2

V. Input arguments

In this section are described the inputs arguments used for the different environments.

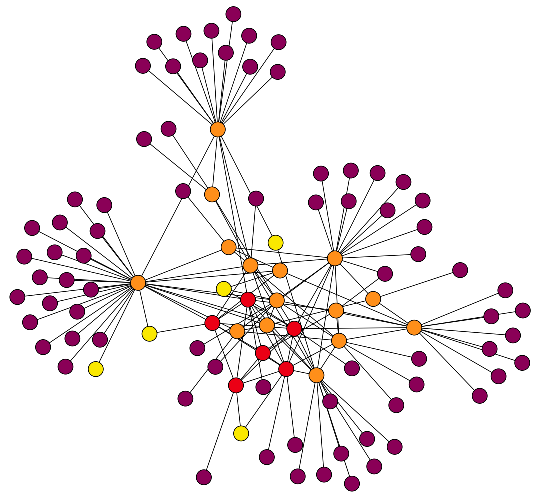
A. Fabrikant arguments

- seeds: 20 different seeds;
- Signaling: "AWA", The input signal determines the messages that the source should send
- Implicit withdraw: active
- Withdraw distributions: 3 different withdraw uniform distributions $[5, 10]$, $[10, 15]$ and $[30, 45]$
- Reannouncement distributions: 3 different announcement uniform distributions $[5, 10]$, $[10, 15]$ and $[30, 45]$
- Processing time: constant with value $0.000\,01$
- Network delay: 3 uniform distributions in $[0.001, 1]$, $[0.5, 3]$ and $[2, 6]$

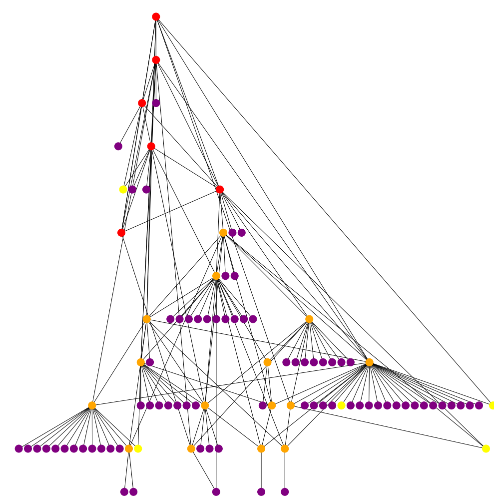
The number of possible different combinations of these values is 540, so for each different MRAI type has been done 540 experiments and in total 2160 expieriments in the fabrikant environment.

B. Clique arguments

- seeds: 10 different seeds;
- Signaling: "AW"
- Implicit withdraw: active
- Withdraw distributions: uniform distribution $[5, 10]$
- Reannouncement distributions: Ininfluent
- Processing time:i uniform distribution $[0.01, 1]$



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



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

Figure 2: Internet like graph colored to show the hierarchicaly structure, 4 tipes of nodes, T (tier 1 mesh), M, CP, C (Customers, purple one)

- Network delay: uniform distribution in $[0.012, 0.1]$

This environment attempt to replicate what has been presented in [3]

In total this environment would run 10 different permutations because the only element that can differ is the input seed. But has been done in a total of 61 experiments changing the MRAI value between 0 and 60, so in total, we had 610 runs, 10 for each MRAI value. It wouldn't have much sense, in my opinion, to run more than one simulation batch per MRAI value, because the repetition of the seed with no difference in any other parameter would have produced the same result.

C. Internet like arguments

- seeds: 10 different seeds;
- Signaling: "A"
- Implicit withdraw: active
- Withdraw distributions: Ininfluent

- Reannouncement distributions: Ininfluent
- Processing time: constant with value $0.000\,01$
- Network delay: uniform distribution in $[0.012, 3]$

The environment has been used for random experiments and fixed mrai experiments.

Like in the clique experiments, in the case of the fixed MRAI, every link had the same timer value. But this time has been used also fractions of seconds to highlight the trend. There have been 121 experiments with MRAI in the ensemble $[0.0, 60]$. In the first fraction $[0.0, 5.0]$ has been used a step of 0.1 doing 51 experiments. The second fraction was $[5.5, 20]$ with a step of 0.5, doing in total 30 experiemnts. The last subset was $[21, 60]$ with a step of 1, doing in total 40 experiments. The final total is 121 experiments and for each of them has been done 10 runs, one for each possible seed of the environment.

The second type of experiments with the internet like environment were run with random graphs. Before running a random experiment the $MRAI_{mean}$ were chosen randomly before the generation of the random graph. In total has been chosen 60 random $MRAI_{mean}$ uniformly distributed in the set $[0, 60]$ the limit 60 has been chosen arbitrarily being the double of the actual standard. 100 random graphs were generated for each $MRAI_{mean}$. Each link would obtain a random value in the set $[0, 240]$ and then all the values would be re-proportioned to respect the $MRAI_{mean}$ At the end for each random graph would be done 10 runs thanks to the 10 different seeds. The total number of this particular configuration is $60 * 100 * 10$ equal to $60\,000$ single runs

VI. Experiments Results

The first results that I would like to examine is the single node results from the fabrikant experiment. In Figures 3 and 4 are represented two signalling outputs of the node number 9.

The x axis represents the number of messages in the output signal. The first y axis, the one on the left, represents the probability to have a certain number of messages in the output sequence and should be used with "withdraw messages", "Advertisement messages" and "Total messages" lines. For example in Figure 3 we can see that there is a really high probability to have 0 withdraws in the output sequence, and we would never see more than 1 withdraw by the fact that the "withdraw line" doesn't go over that value of the x axis. In the same way, we can read the other lines, for example in the same figure is possible to see that the more probable output signal is composed by 3 messages, because is the highest point of the blue line. The second y axis, the one on the right, represents the number of **unique** states. This axis should be used with the line that represents the "possible outputs". Looking the Figure 3 is possible to see that for output signals with 6 messages we have more than 40 uniques output states.

Knowing that the output signalling is strictly dependent by the input that a node receives and the evaluation time of those messages we can already see the effects of an inconvenient MRAI setting. In Figure 4 there are a lot more output states. In those experiments we reach event output states of length 16 and the probability to have at least one withdraw is higher to

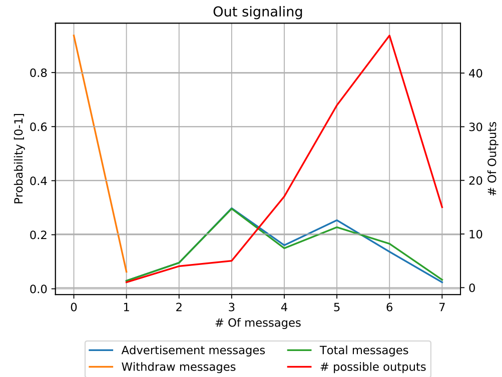


Figure 3: Fabrikant MRAI fixed 30 seconds, node 9 signalling output

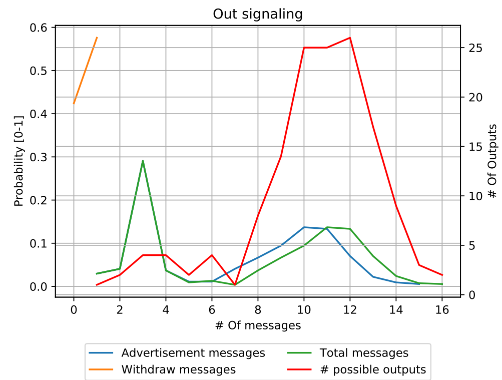


Figure 4: Fabrikant MRAI descendent, node 9 signalling output

the probability to don't have one. Knowing that the implicit withdraw system is active having one withdraw means that the node has no other possibilities to withdraw the network not knowing any other path.

Like we said before we used 4 different MRAIs strategies in the fabrikant environment, and them are compared in the Figures 5 and 6. The no mrai method is strictly dependent on delays and other events of the network and is possible to see that it has the smallest convergence time but with the higher number of messages necessary to reach the convergence. The 30 seconds strategy could be the slowest one because if something goes wrong is necessary to wait a long time to repair the damage, but it wouldn't require too many messages on the other side. The descendant method seems a good solution on the convergence time side, but on the other side, like is described in [1] it could easily lead to a lot of messages.

In Figures 7 and 9 are reported the general network results obtained in the clique environment. The goal of this study is to see in a general way how MRAI could influence even a small network as this clique of 15 nodes.

In Figure 7 every point of the plot is the mean of the 10 runs executed with a fixed MRAI. On the x axis is represented the number of messages correlated with the convergence time on the y axis. The red points are the Pareto front of the set of all points. We can see from the plot that a lot of experiments has a mean of messages sent around 2000 independently from the MRAI so we can guess that after a threshold of MRAI the number of messages stabilizes around that value. On the

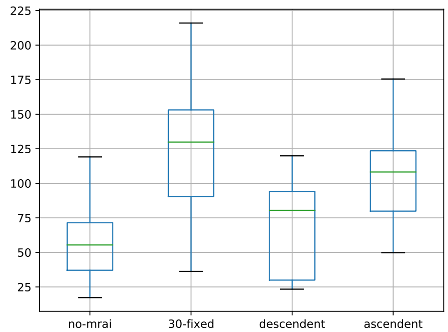


Figure 5: Fabrikant experiments convergence time comparison

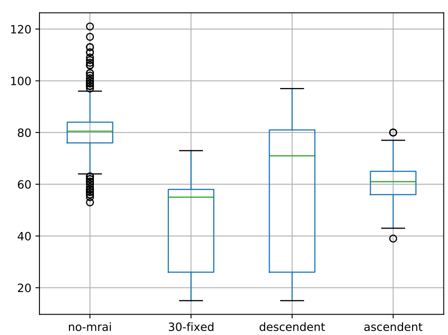


Figure 6: Fabrikant experiments number of messages comparison

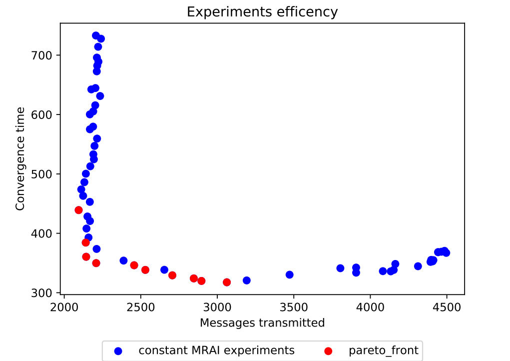
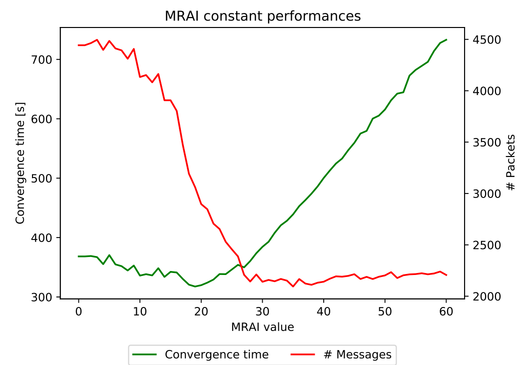
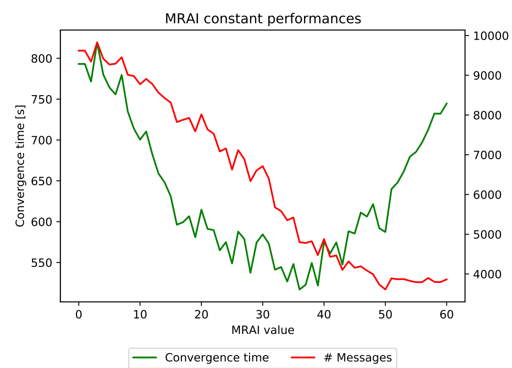


Figure 7: Pareto front in the clique environments



(a) Clique environment with Implicit Withdraw active



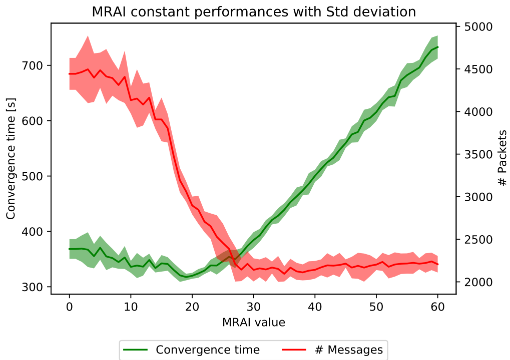
(b) Clique environment **without** Implicit Withdraw active

Figure 8: Evolution of the number of messages sent and the convergence time as MRAI grows in the clique environment

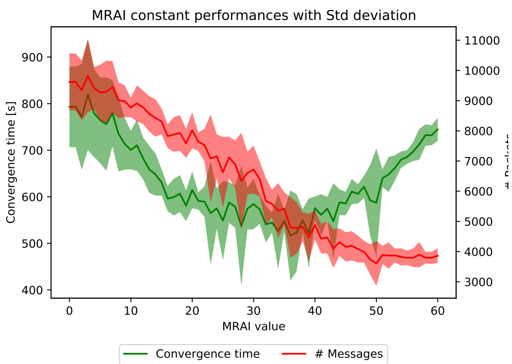
other side, before this threshold we can guess there is a lot of variance in the number of messages but the mean convergence time is similar. These guesses are confirmed by the Figure 9 where we can clearly see those trends. The two y axis are used to represent the trend of the convergence time and the number of messages transmitted in relation of the MRAI value. Error bars represent the standard deviation of the time/messages for the 10 experiments that represent each point. Interesting the fact that the standard deviation becomes smaller icreasing the MRAI timer. The two **????** refers to two different situations in the same clique graph. In the first figure the implicit withdraw is active, in the second has been deactivated, the difference is evident, the number of messages, the convergence time and even the standard deviation is higher without the implicit withdraw active. This confirm what has been presented in [4]

The next experiment that I would like to study is the fixed MRAI strategy on the internet like environment. The two Figures 10 and 11 have the same structure of the clique results but this time we can see a lot fewer messages transmitted to reach the convergence. Also this time we can see in Figure 10 that a lot of experiments are concentrated in the range between 320 and 340 messages with a high variance on the convergence time. In fact, we can see from Figure 11 that the trend is similar to the one that we saw in the clique graph, but this time the "transmitted messages" line has a steep fall, it reaches the constant state in few seconds (the clique experiment in Figure 9 took more than 20 seconds). But we can say the same thing for the convergence time too. Tose results are similar to the one available in [3]. The standard deviation has a different evolution, is possible to see that it doesn't becomes lower but remains constant. This is caused by the fact that in this network the MRAI is more influent than the delay that at each step becomes less influent.

Some more general results could be saw taking into consideration the random MRAI strategy on the internet like graph. The results are visible in Figure 17. The first thing that we can see is that this time there are not huge spikes for certain messages amount. The number of messages sent is more distributed between 300 and 360. Also, the convergence time is more



(a) Clique environment with Implicit Withdraw active



(b) Clique environment **without** Implicit Withdraw active

Figure 9: Evolution of the number of messages sent and the convergence time as MRAI grows in the clique environment, the range around the line represent the standard deviation of the experiment

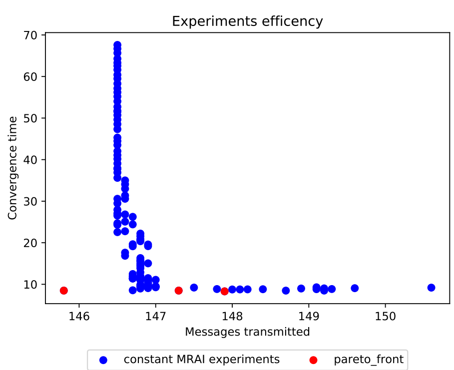
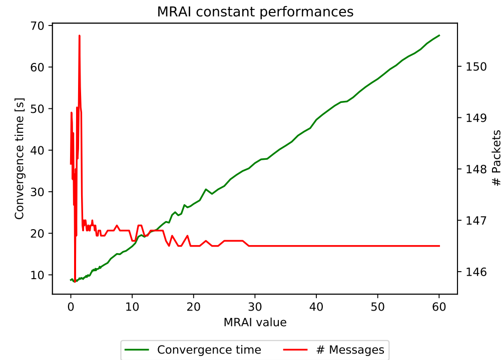


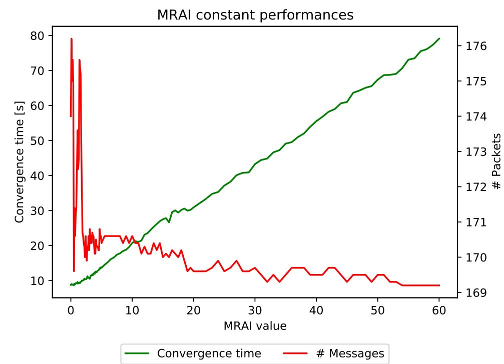
Figure 10: Pareto front in the Internet like constant MRAI environment

distributed, thanks to the random MRAI distribution. We can guess that if a central node has a huge MRAI timer and it transmits incorrect information it would act as a bottleneck for the update with the correct information.

From the comparison of the two MRAI strategies we can see in Figure 18 that the constant MRAI cover a really small part of the random strategy. And for small values of MRAI is also possible for the constant strategy to send more messages than the random one.



(a) Internet like constant MRAI environment with Implicit Withdraw active



(b) Internet like constant MRAI environment **without** Implicit Withdraw active

Figure 11: Evolution of the number of messages sent and the convergence time as MRAI grows in the **internet like** constant MRAI environment

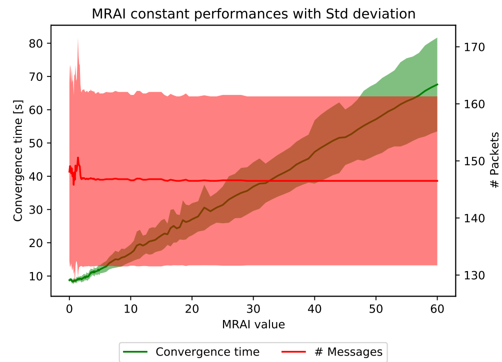
References

[1] A. Fabrikant, U. Syed, and J. Rexford, "There's something about MRAI: Timing diversity can exponentially worsen BGP convergence," in *30th IEEE Int. Conf. on Computer Communications (INFOCOM 2011)*, Shanghai, China, Apr. 2011.

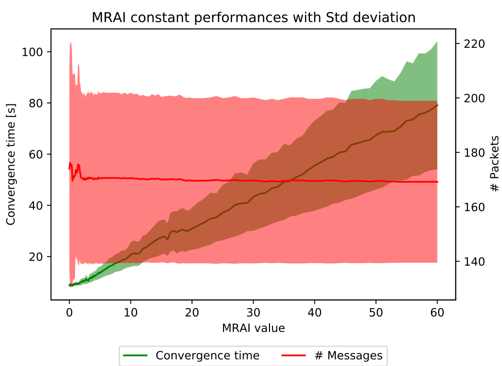
[2] A. Elmokashfi, A. Kvalbein, and C. Dovrolis, "On the Scalability of BGP: The Role of Topology Growth," *IEEE Jou. on Selected Areas in Communications (JSAC)*, vol. 28, no. 8, pp. 1250–1261, 10 2010.

[3] 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.

[4] 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.



(a) Internet like constant MRAI environment with Implicit Withdraw active, standard deviation



(b) Internet like constant MRAI environment **without** Implicit Withdraw active, standard deviation

Figure 12: Evolution of the number of messages sent and the convergence time as MRAI grows in the **internet like** constant MRAI environment. The range around the lines represent the standard deviation of the experiment

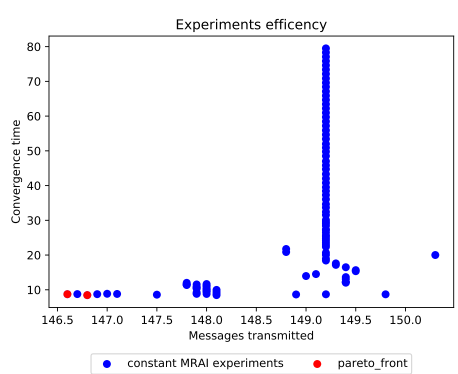
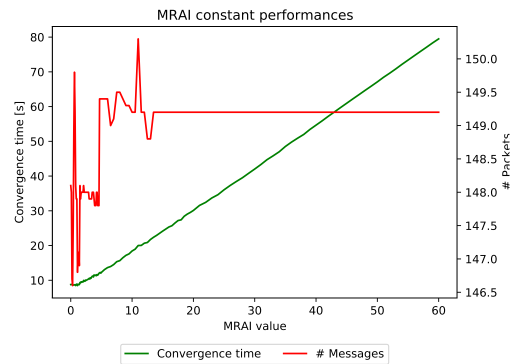
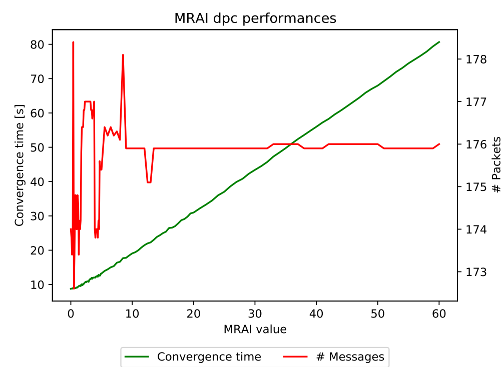


Figure 13: Pareto front in the Internet like DPC MRAI environment

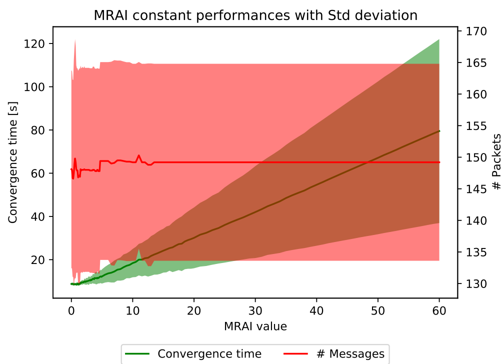


(a) Internet like DPC MRAI environment with Implicit Withdraw active

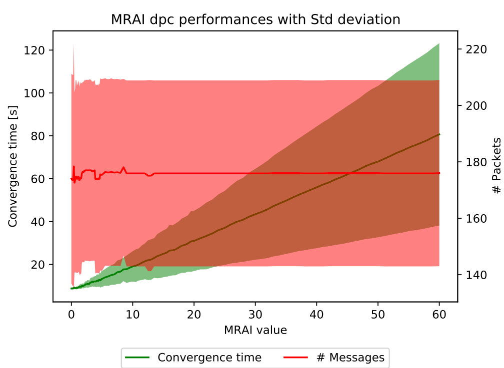


(b) Internet like DPC MRAI environment **without** Implicit Withdraw active

Figure 14: Evolution of the number of messages sent and the convergence time as MRAI grows in the **internet like** constant MRAI environment

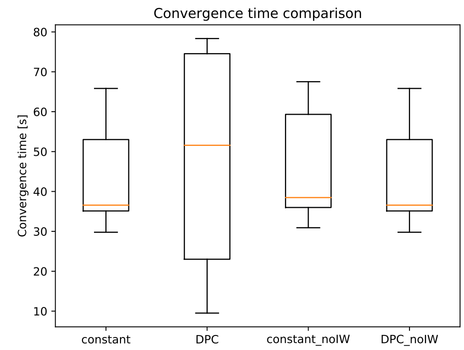


(a) Internet like DPC MRAI environment with Implicit Withdraw active, standard deviation

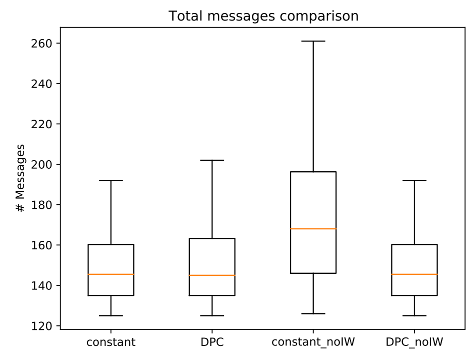


(b) Internet like DPC MRAI environment **without** Implicit Withdraw active, standard deviation

Figure 15: Evolution of the number of messages sent and the convergence time as MRAI grows in the **internet like** DPC MRAI environment. The range around the lines represent the standard deviation of the experiment



(a) Convergence time comparison between different environments



(b) Messages necessary to reach the convergence comparison

Figure 16: Comparison between the DPC MRAI strategy and the constant MRAI strategy, 100 experiments with the same conditions with $MRAI_{mean}$ equals to 30.0

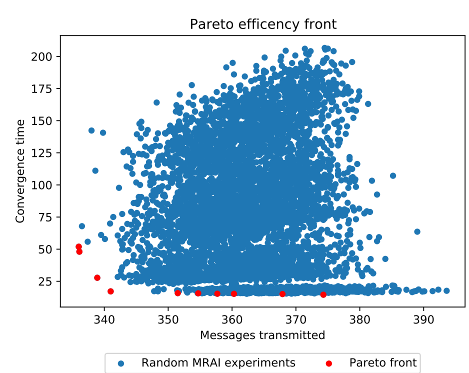


Figure 17: Pareto front in the Internet like random environment

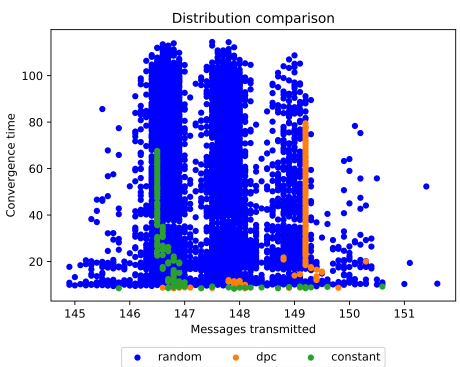


Figure 18: Comparison of the random strategy and the constant one in the internet like environment