SPAM Detection Server Model Inspired by the *Dionaea Muscipula* Closure Mechanism: An Alternative Approach for Natural Computing Challenges

Rodrigo Arthur de Souza Pereira Lopes[1], Lia Carrari R. Lopes[2],
and Pollyana Notargiacomo Mustaro[1]

[1] Universidade Presbiterian a Mackenzie, Programa de Pós-Grad. em Engenharia Elétrica
Rua Piauí, 130 – Prédio Modesto Carvalhosa, Anexo do 2 º Subsolo, São Paulo, Brazil
[2] Instituto de Pesquisa em Tecnologia e Inovação
Av. São Luis, 86 cj 192, São Paulo, Brazil
rodlopes@gmail.com, lia.carrari@ipti.org.br,
polly@mackenzie.br

Abstract. Natural computing has been an increasingly evolving field in the last few years. Focusing on the interesting behaviours offered by nature and biological processes, this work intends to apply the metaphor of the carnivorous plant "*Dionaea muscipula*" as a complementary defence system against a recurring problem regarding internet and e-mails: spam. The metaphor model presents relevant aspects for further implementation and debate.

Keywords: Natural computing; *Dionaea muscipula*; spam.

1 Introduction

Natural Computing has been an increasingly evolving field. By using nature as its inspiration, it has provided unexpectedly efficient solutions to problems that were before unsolvable. Issues concerning this area are inconstant discussion in the scientific community. In that sense, computer scientists are uniting technical and computational researches with biology specialists to achieve new sources for solutions and innovation. Originating ideas for methods and algorithms as well as applications in natural computing stimulates creativity and multidisciplinary research.

This work intends to take advantage of such characteristics to propose a parallel solution to the issue of internet spam, based on the closure mechanism of a carnivorous plant. Here, the theoretical research and models on the theme will be explored, inspiring future implementations and modifications on current systems. It may also be used on similar situations to avoid not only spam mails, but also repeated actions from bots and other sources with malicious intents towards a particular server. For this situation, one could see the OPAALS request for service building system as a potential target.

F.A. Basile Colugnati et al. (Eds.): OPAALS 2010, LNICST 67, pp. 192–198, 2010.

This work is divided in four parts. Firstly, a brief overview of the different branches and applications of Natural Computing. Secondly, an explanation about the closure mechanism of the *Dionaea Muscipula*. Next, a description of the proposed problem and its encircling subject – spam. Lastly, the application of the metaphor in the three different proposed levels, followed by the conclusion.

2 Natural Computing

Natural computing is a discipline that bases itself on the study, interpretation and reproduction of natural phenomena. This area of study produces methods, algorithms and approaches to solving problems that are computationally untreatable (problems that do not have an optimal solution under a polynomial time algorithm and belong to NP-complete or NP -hard classes [2], and proposes innovative, novel solutions to problems that already have efficient solutions [3].

It currently focuses on three different branches, each of them having their own approaches and methods for tackling different kinds of problems. Firstly, there is nature-inspired computing. This branch makes use of nature and natural phenomena such as the organization of an ant colony, the inner workings of the immunological system, or the human brain to propose algorithms capable of solving problems. As an example, today there is a rather extensive use of Neural Networks for pattern recognition and other similar problems that are not easily solved by conventional means.

Secondly, natural computing may be used to simulate the natural phenomena mentioned in the previous paragraph, like simulating the flow of a river, the collective behaviour of a school of fish, or even drawing the shape of a mountain. This area presents diverse techniques for modelling patterns and structures, such as Cellular Automata and Artificial Life.

Lastly, the third branch concerns computing with natural materials - more specifically, computing with materials other thansilicon. Moore's law predicts that the number of transistors in a chip doubles every two years [4] - this has held true until now, and is expected to hold true for at least an other two decades . Should that be the case, chips will very soon reach their peak miniaturization levels, and thus, peak processing power. From this situation the need to find new materials arises, and that is the task undertaken by this branch of natural computing.

One of the most popular methods originated by the natural computing is the Genetic Algorithm. The Genetic Algorithms are mainly used in search and optimization obtaining solutions though the evolution of populations of encoded feasible solutions (individuals) [5]. Then, the population is updated by mimicking the natural evolution mechanisms such as selection and reproduction processes. The algorithm also takes into consideration reproductive aspects, such as crossover and mutation.

Following that idea, this work proposes a nature inspired model that will focus on the first described branch, thus employing the metaphor to the proposed problem. To do so, one must understand the workings of the chosen metaphor, the closure mechanism of the *Dionaea Muscipula*.

3 Dionaea Muscipula

The carnivorous plant *Dionaea muscipula*, popularly known as Venus flytrap (Fig. 1), as been studied for over a century, mainly due to its rapid closure mechanism [6]. It is a species of carnivorous plant that captures its prey by rapidly trapping them between its leaves [7], in the order of one tenth of a second.

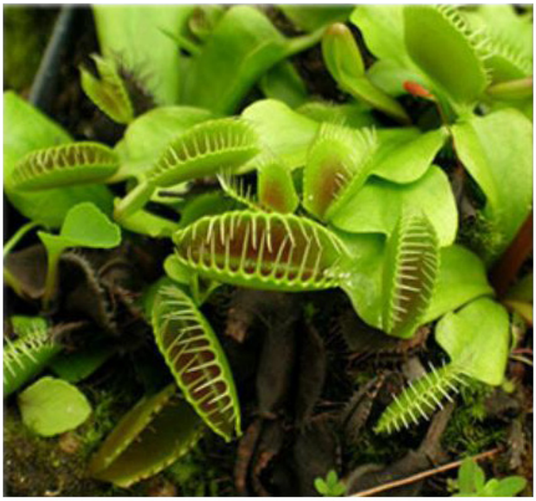

Fig. 1. Venus Flytrap (*Dionaea muscipula*) [8]

The plant consists of several leaves, each one divided into two parts that are held together by a midrib. These parts are called lobes, and have at least three sensitive hair triggers positioned around their surface (Fig. 2), which are covered with a red pigment which attracts insects [7].

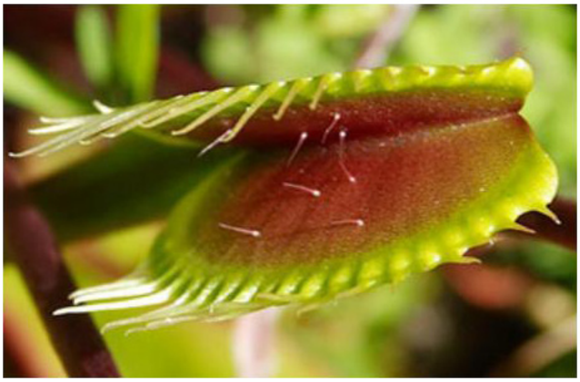

Fig. 2. Sensitive hair triggers from the Venus Flytrap [8]

Whenever one of those hairs suffers stimulation – that is – once an insect or another creature touches it, an electrical signal is released, generating action potential that propagates across the lobes and stimulates its cells. What stimuli are generated is still under debate, but there are two generally accepted mechanisms, which may simultaneously play a part in the closure of the lobes.

The first one involves the moving of hydrogenions causing swelling by osmosis, and thus changing the shape of the lobe; the second one involves the formation of water by osmosis causing the collapse of the lobe cells [8]. But these are very simplistic descriptions and a detailed description of chemical reactions and mechanisms of this closure are not relevant to the development of this work.

Under normal temperatures, two stimuli are required to trigger the closure mechanism; however at high temperatures, close to 40° C, only one stimulus is sufficient. After trapping the prey (Fig. 3), further stimuli from the prey's thrashing about will tighten the grip of the mechanism, eventually closing the trap hermetically [6].

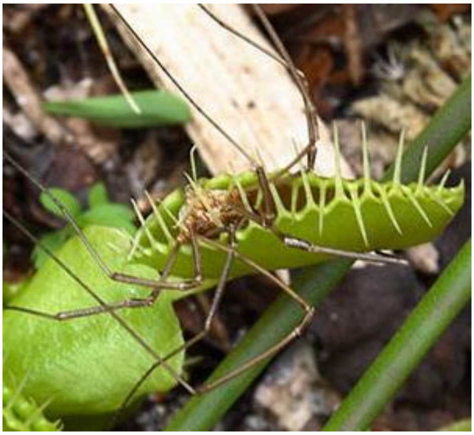

Fig. 3. Closed Venus Flytrap and its prey [8]

Through the abstraction of the mechanisms of the Venus Flytrap, this work constructs a metaphor, applying its methods to Internet spam e-mail identification. Considering that there are many different approaches for this problem, some ideas for spam definition and recognition will be explored next.

4 SPAM

Spam is an unsolicited message that is delivered to a user's mailbox. Generally, it contains some sort of advertising information about services or products, but it also may contain fraud attempts or harmful content, such as advertising false bank sites that contain trojans (malicious code that could steal personal information or offer access to one's computer), chain mails, or pirated software offerings [10].

Several ways of combating spam have been proposed over the years, such as filtering by content, blocking SMTP servers, greylisting and others, but this work will focus on the complementary solution (as it is not a stand -alone defence against spam, but one of many) proposed by Lieven, Scheuermann, Stini and Mauve [11].

The SMTP RFC [12] specifies that upon receiving a temporary error, a SMTP server should wait for a period of time before trying to re-send the message, and should not try to deliver other messages to that same domain during this wait time. A *spambot* (an automated spam-sending script or software) or spam agent (infected machine, script, etc.), however, focuses on throughput, meaning that it will repeatedly try to deliver a message in spite of receiving a temporary error [11].

According to [11], effective results could be obtained by observing such retry patterns. Basically, spammers are not RFC compliant and will try at every cost to deliver their messages. By observing the retry intervals, Lieven et al. was able to filter the spammers based on the frequency of their requests. One of the steps used in the proposed method was to whitelist (mark as trustworthy) all mail hosts that were the destination of mails sent by local hosts.

This potential flaw allows infected machines to utilize the whitelisted mail host to convey unwanted spam to the local users, and this is one of the places the metaphor comes in.

5 Metaphor Application

This work intends to apply the metaphor of the closure mechanism of the *Dionaea muscipula* to computational problems, mostly focusing on spam. This will be done in three different levels. Firstly, by "trapping" spammers after a predetermined number of stimuli has been reached; secondly, by effectively banning the spammers once a time interval is elapsed after the trap with repeated stimuli; and finally, by implementing an effective "heat –zone", when total numbers of stimuli are enough to compromise actual CPU load.

The potential flaw described in the previous chapter allows an infected machine to use the trusted mail hosts to deliver spam to local users. The change proposed in this work is to not whitelist those hosts immediately, but apply different types of observance on three different levels.

The *Dionaea muscipula* closes its lobes capturing its prey after two stimuli in a given interval of time. Therefore, it is proposed that not one central spam barrier (a proxy to stand before the actual SMTP server) be implemented, but several, each covering an array of different delivery addresses.

As a simplistic example of the metaphor, *spam barrier* I may cover addresses starting with A and B, while *spam barrier* II covers addresses starting with B and C, and so on. Other, better algorithms for covering e-mails, may and should be applied, executing actual load -balancing between each spam barrier but not compromising the final purpose of this spread. Each *spam barrier* keeps track of its blacklisted addresses (the ones which keep trying repeatedly to deliver the message), and this information should be synchronized at predetermined intervals of time. The separation of addresses by starting letters confers additional "proof" that the same host is sending mail to multiple addresses, further insuring that it is a suspicious address.

In the same way that the *Dionaea muscipula* closes its lobes after two stimuli in an interval of time, this system may stipulate a number of stimuli to block the e-mails from a given host for a period of time.

Each occurrence of spamming on a *spam barrier*'s record is then considered a stimulus; the synchronization process is responsible for taking into account each stimulus, and calculating the total number of stimuli received from each remote host to decide on an action course. This process is also assigned the task of keeping a reduced database of "totals", which indicate the total times a given host has been flagged as a spammer; these totals are to be considered part of the metaphor in what relates to additional stimuli, the second level of protection. Repeated stimuli after each blocking period will eventually hermetically close the trap – that is, impose a much larger period of blocking, or eventually even applying a permanent ban to such host based on its previous history.

After the blocking period, the host may be unblocked again (barring the previous issuing of a permanent ban), in the event that the occurrence was perceived as being related to an infected machine, and not a legitimate spamming source (an actual server dedicated to sending spam). This situation could be detected over a period of time, since infected machines tend to be cleansed of its viruses and Trojans periodically, so a certain host would send spam intermittently. That means it would be off the blocking period for larger timeframes than it would be blocked.

This step is related to the *Dionaea muscipula's* capacity to identify if the agent responsible for stimuli is a potential prey and not something else like a rain drop. Even though the stimulation happens and the mechanism is triggered a combination of factors (electrical and chemical reactions) is considered before "processing" the victim.

The third and final step of applying the metaphor is the temperature. The absolute amount of incoming delivery requests is considered equivalent to a certain temperature, where a small amount equates to a small temperature, and a high amount to a high temperature. Analogously to the *Dionaea muscipula*, in a high temperature environment, a reduced number of stimuli could be assigned to trigger the closure mechanism. This could help alleviate the CPU load compromised by a large amount of delivery requests coming from different directions, effectively filtering spam more immediately, and redirecting the CPU power to legitimate requests.

In the case of a more generic situation, such as a request spamming bot, the same system could be used. This could be illustrated by using the request for service system used in OPAALS, as mentioned before [1].

This system uses a requirement from the user interface to generate a query and combine atomic services in order to build a more complex variant, in which the user requirements should be met. For this task, valid combinations of resources are generated through the querying of bounded resources. Resources that are unbounded are then repeatedly queried, once for each of the previously generated valid combinations. These may generate the final combinations that compose a whole service [1].

This operation may be exploited by a user (in this case a bot) submitting requirements repeatedly and taking advantage of the processor-heavy queries to overload the server. The same mechanisms would apply in this situation.

6 Conclusion and Further Work

Albeit not technically efficient due to implementing several *spam barriers* instead of a centralized unit, this mechanism could prove useful in offering more accurate detection of spam agents. It also reduces CPU overload, consequently compromising the system less than other applications. The abstract implementation of the metaphor could be applied in three different levels, possibly indicating it as a valid alternative or complement to current spam solutions.

Implementation details and a deeper exploration of the chemical reactions as specific algorithms for detection and parameters can be studied in the future. A broader metaphor may also be suggested, considering the *Dionaea muscipula*'s biological environment applied to the system architecture.

References

1. Krause, P., Marinos, A., Moschoyiannis, S., Razavi, A., Zheng, Y., Kurtz, T., Streeter, M.P., Gabaldón, J.E.: Deliverable D3.3: Full Architecture Definition. In: OPAALS (July 2008), http://files.opaals.org/OPAALS/Year_2_Deliverables/WP03/D3.3.pdf
2. Ascia, G., Catania, V., Palesi, M., Parlato, A.: A Evolutionary Approach for Reducing the Energy in Address Buses. In: Proceedings of the 1st international Symposium on information and Communication Technologies. Ireland, ACM International Conference Proceeding Series, vol. 49, pp. 76–81 (September 2003)
3. de Castro, L.N.: Fundamentals of natural computing: basic concepts, algorithms, and applications. Chapman Hall, Boca Raton (2006)
4. Intel. Moore's Law: Made real by Intel innovation. Intel Corporation (February 2008), http://www.intel.com/technology/mooreslaw/
5. Holland, J.H.: Adaptation in Natural and Artificial Systems. Univ. of Michigan Press, Ann Arbor (1975)
6. Volkov, A.G., Adesina, T., Jovanov, E.: Closing of Venus Flytrap by Electrical Stimulation of Motor Cells. Plant Signal Behav. 2(3), 139–145 (May-June 2007), http://www.ncbi.nlm.nih.gov/pmc/articles/PMC2634039/
7. Forterre, Y., Skotheim, J.M., Dumais, J., Mahadevan, L.: How the Venus flytrap snaps. Letters to Nature 433, 421–425 (2005)
8. Botanical Society of America.: Venus Flytrap - Dionaea muscipula - Carnivorous Plants Online - Botanical Society of America (2009), http://www.botany.org/carnivorous_plants/venus_flytrap.php
9. Markin, V.S., Volkov, A.G., Jovanov, E.: Mechanism of trap closure by Dionaea muscipula Ellis. Plant Signal Behav. 3(10), 778–783 (October 2008), http://www.ncbi.nlm.nih.gov/pmc/articles/PMC2637513/
10. Indiana University: What is spam? Knowledge Base (2009), http://kb.iu.edu/data/afne.html
11. Lieven, P., Scheuermann, B., Stini, M., Mauve, M.: Filtering Spam Email Based on Retry Patterns. In: IEEE International Conference on Communications ICC '07, pp. 1515–1520 (June 2007)
12. Postel, J.B.: Simple Mail Transfer Protocol. RFC 2821, http://www.ietf.org/rfc/rfc2821.txt (April 2001)