



UNIFOR

Improving Load, Performance and Stress Evolutionary Testing using a Hybrid Metaheuristic Approach

IAdapter

Nauber Gois

naubergois@gmail.com
<http://www.unifor.br/>

October 20, 2015

Improving Load, Performance and Stress Evolutionary Testing using a Hybrid Metaheuristic Approach

Fraancisco Nauber Bernardo Gois

Universidade de Fortaleza

20 de Outubro de 2015

Sumário

- 1 Advisors
- 2 Introduction
- 3 Systematic Review
- 4 Load, Performance and Stress Tests
- 5 Types of Workloads
- 6 Evolutionary Tests
- 7 Hybrid Metaheuristic
- 8 Search Based Tests
- 9 IAdapter
- 10 Experiments
- 11 Schedule
- 12 Conclusion
- 13 References

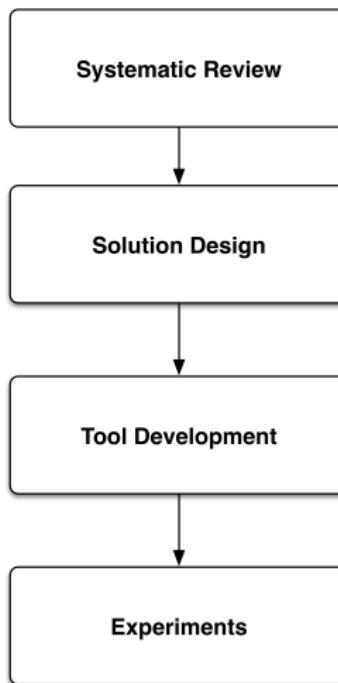
Advisors

- Advisor: Pedro Porfírio Muniz de Farias
- Co-Advisor: André Luís Vasconcelos Coelho

Introduction

The purpose of this research is to propose and investigate the pros and cons of a novel hybrid metaheuristic approach using Genetic Algorithms, Simulated Annealing and Tabu Search to automatically perform load, performance and stress testing.

Research Activities



Bibliography Search Strategy

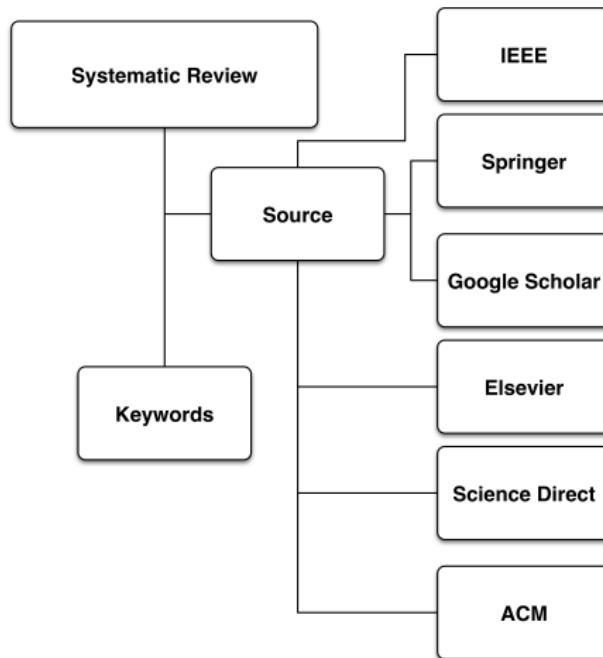
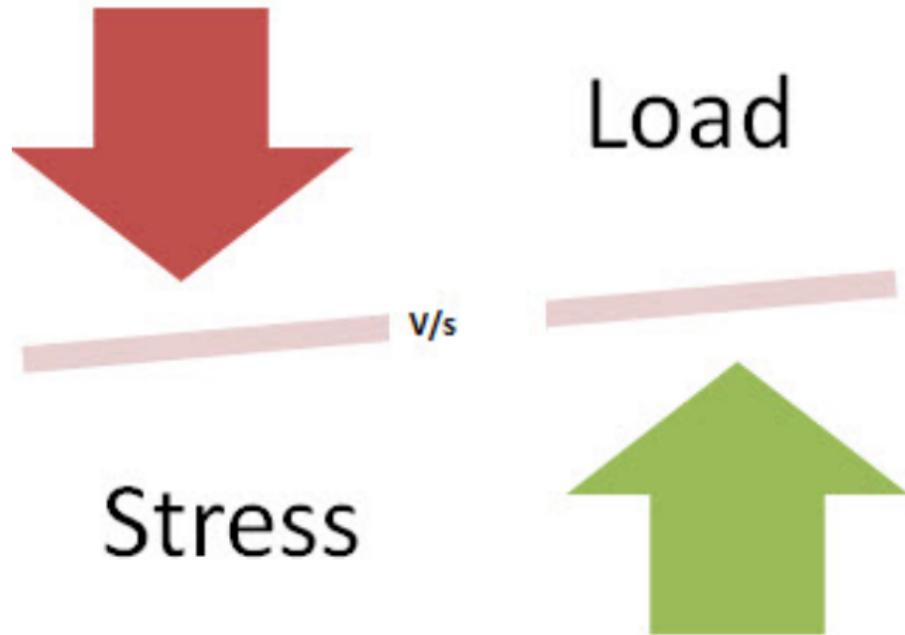


Figura : Search Strategy

Search terms:

- Stress Testing: Search-based Testing, Genetic Algorithms, Stress Testing, Test Tools, Test Automation, Empirical Analysis, Denial of Service, Ramp-Up time, Think Timer, Response Time, Bandwidth Throttle, Dynamic Stress Testing, Evolutionary, Heuristic, Search-Based, Metaheuristic. optimization, genetic algorithms, genetic programming.

- Performance Testing: Performance Testing, Web-based Systems, Software Testing, Model-Based Testing, Software Product Line, Regression Testing, Test Failure Prediction, Genetic Metric Selection.
- Load Testing: Markov chain, Automatic Test Case Generation Algorithms, Domain-based reliability measure, Fault detection, Load Test suites, load testing, Reliability, Resource allocation mechanisms, Software testing, System degradation.



Load, Performance and Stress Tests

The Performance Test aims at verifying a specified system performance. This kind of test is executed by simulating the access of hundreds or more simultaneous users over a defined time interval. The purpose of this test is to demonstrate that the system meets its performance objectives [Di Lucca and Fasolino, 2006][Sandler et al., 2004].

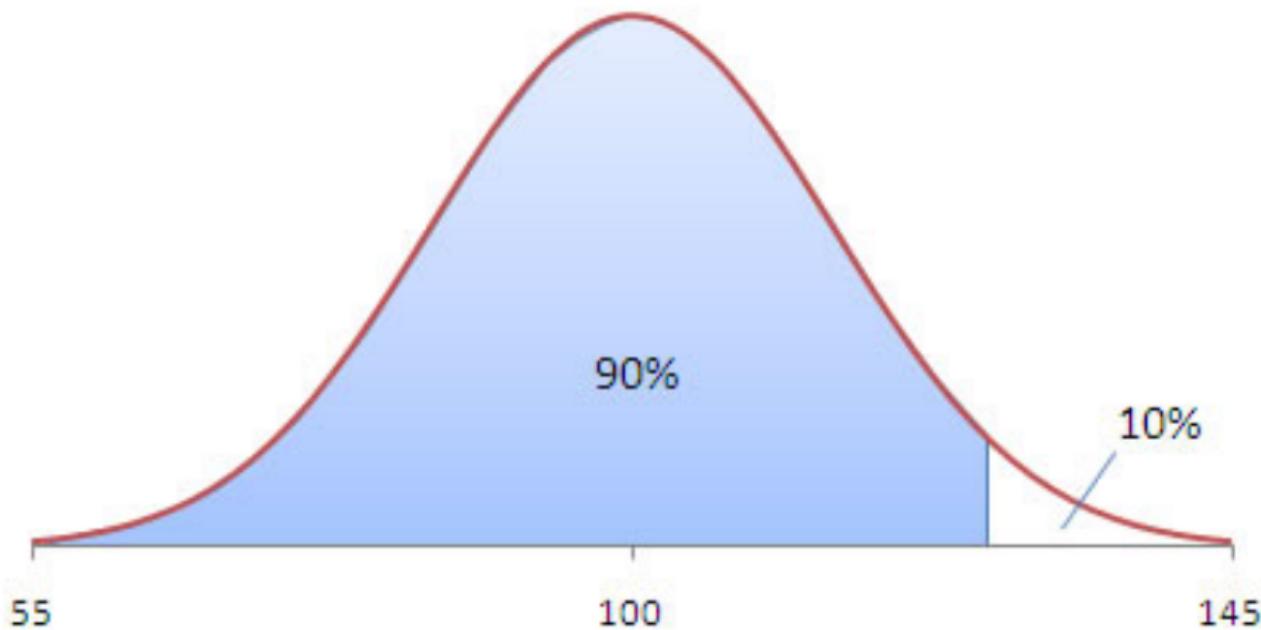
Load, Performance and Stress Tests

In load tests, the system is evaluated in pre-defined load levels. The aim of this test is to reach the performance targets for availability, concurrency, throughput and response time of the system. Load Test is the closest to real application use [Molyneaux, 2009].

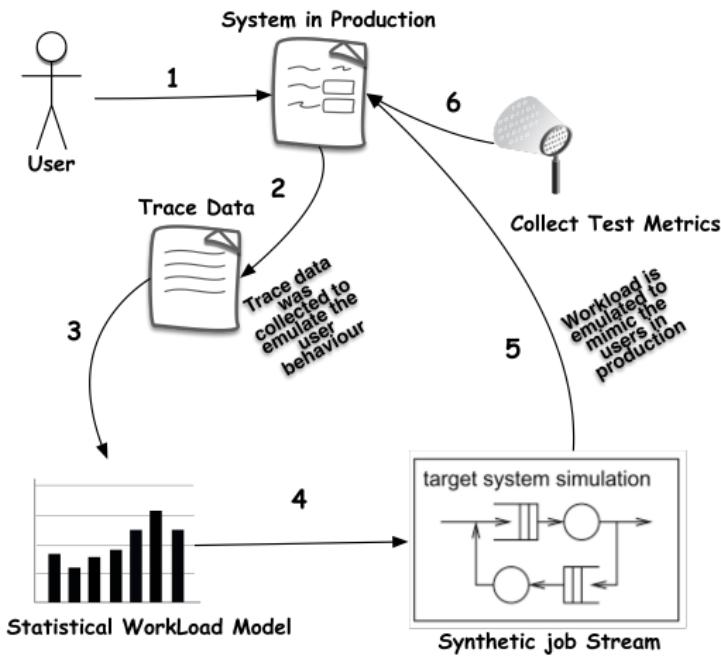
Load, Performance and Stress Tests

Stress test verifies the system behaviour against heavy workloads, being executed to evaluate a system beyond its limits, validate system response in activity peaks and verify whether the system is capable of recovering from these conditions [Sandler et al., 2004].

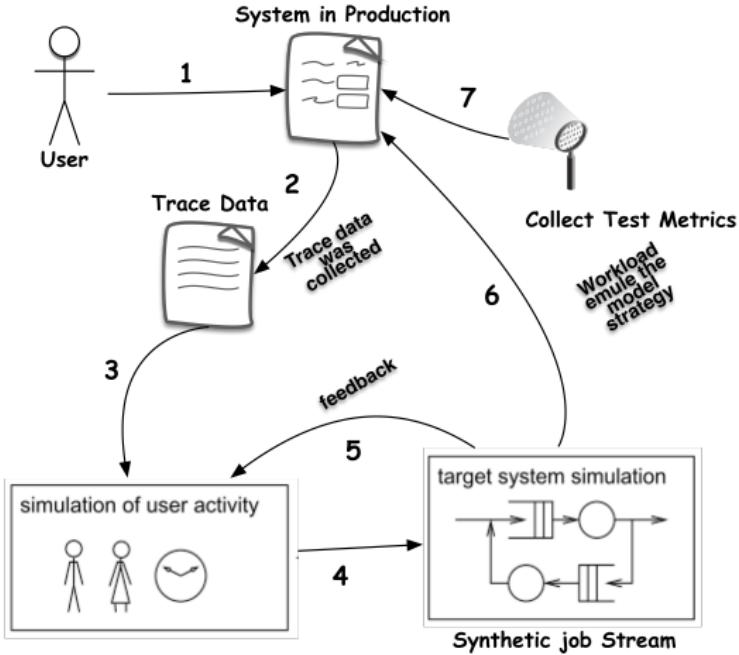
Load, Performance and Stress Tests-90 Percentile Line



Descriptive WorkLoad



Generative WorkLoad



Evolutionary Tests

- The main objective of evolutionary testing in performance, stress and load tests is to find test scenarios which produce execution times violating the timing constraints specified. If a temporal error is found, the test was successful [Sullivan et al.,].
- The application of evolutionary algorithms to load, performance and stress tests involves finding the best and worst case execution times (BCET, WCET) to determine if timing constraints are fulfilled [Afz, 2009].

Hybrid Metaheuristic

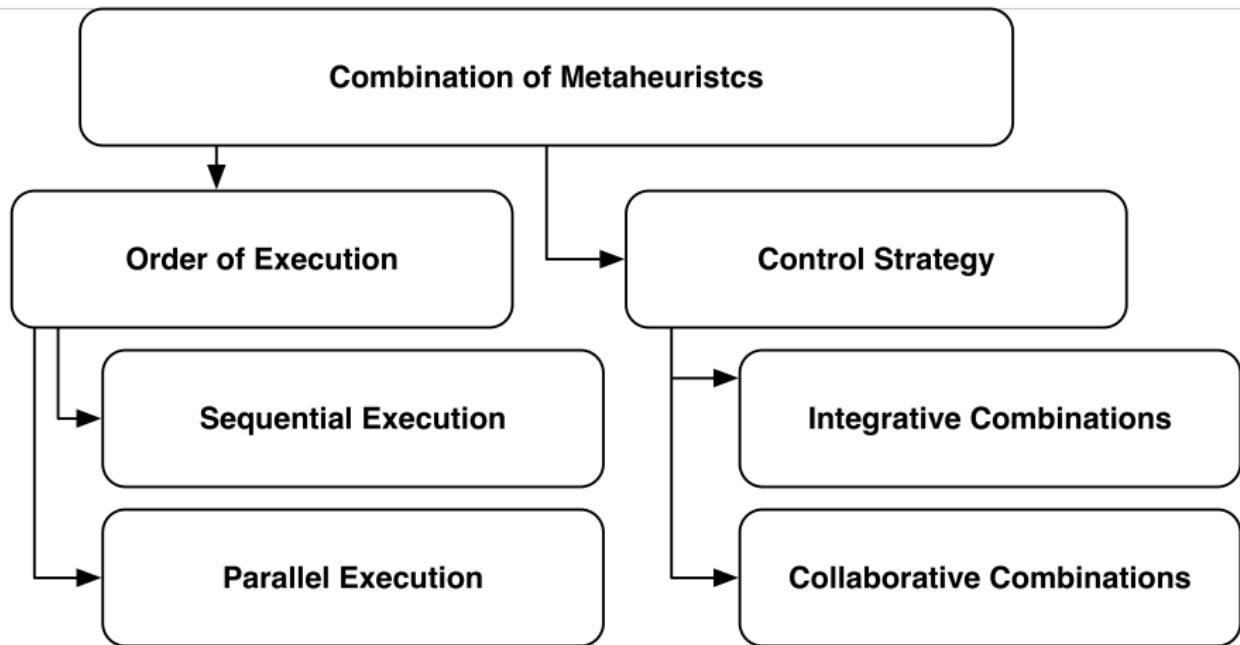


Figura : Categories of metaheuristic combinations

Search Based Tests-Afzal Study

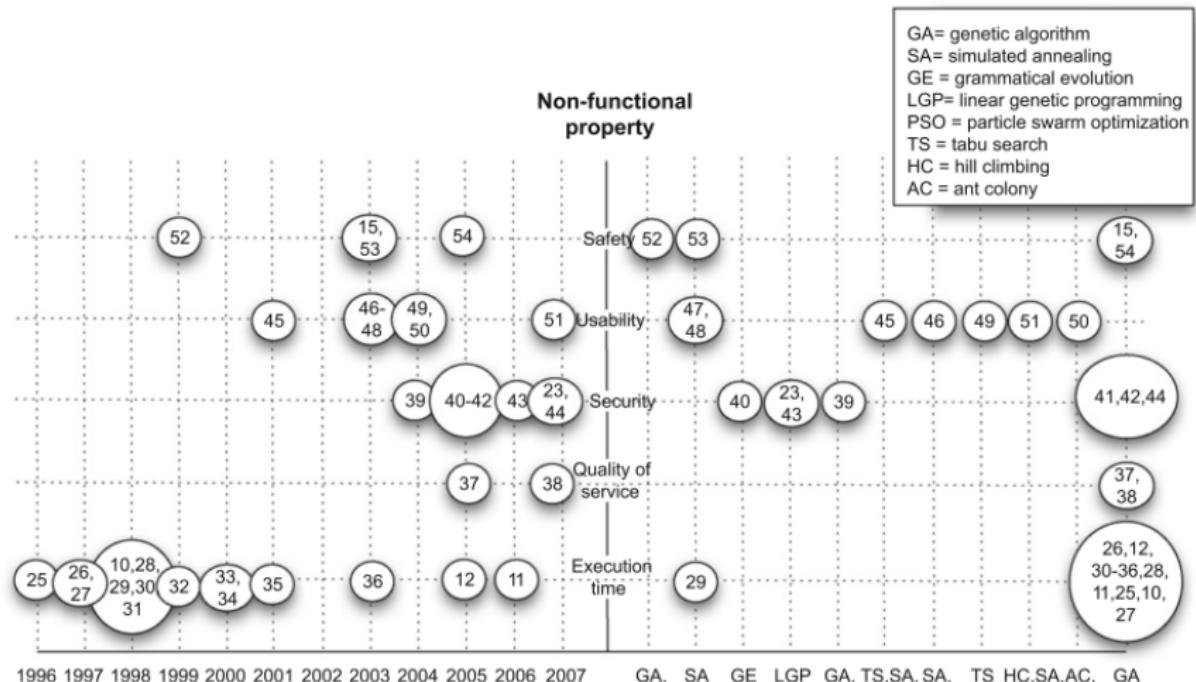
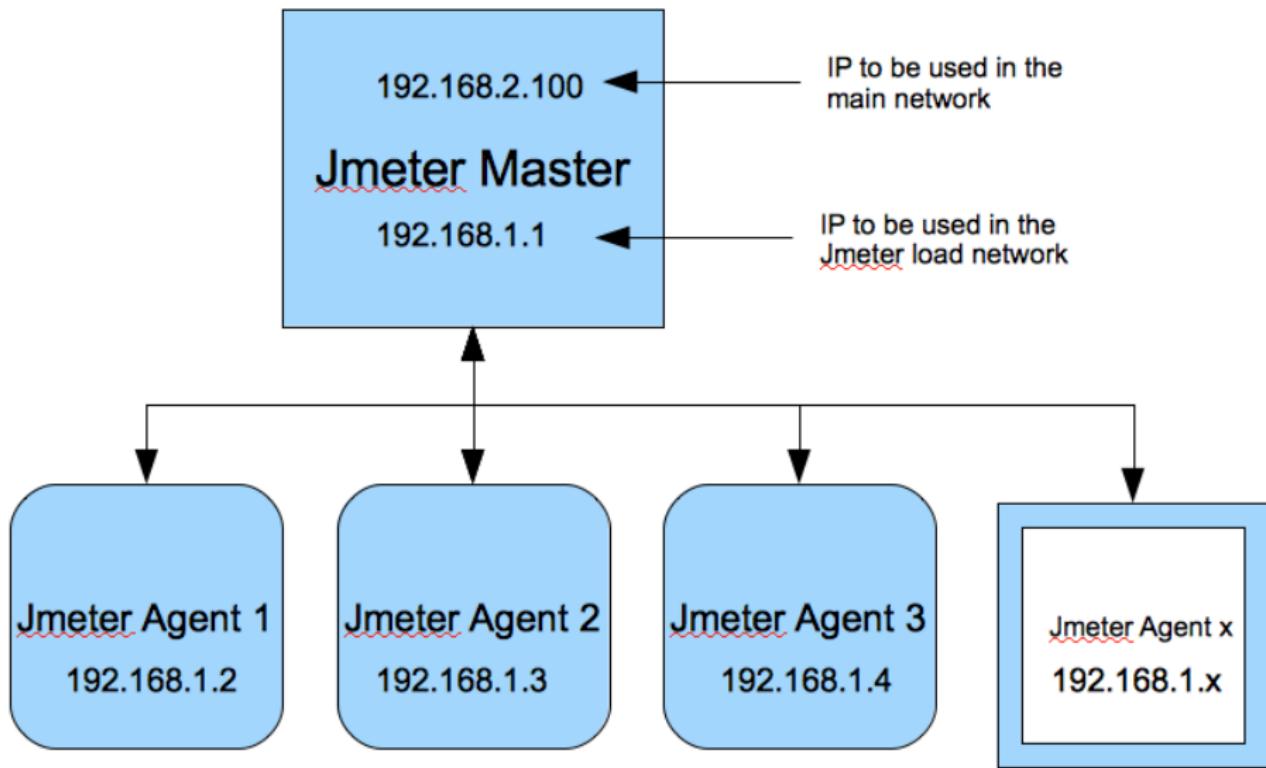


Figura : Distribution of NFSBT research over range of applied metaheuristics and time period [Afz, 2009]

Search Based Load, Performance and Stress Tests

Hybrid Metaheuristic	Prototypes		Functional Tool
	Execution Time	Processor Cycles	Execution Time
			IADAPTER Gois, 2015
GA	Alander, 1996 e 1998 Sullivan, 1998 Wegener, 1997 Briand, 2005 Canfora, 2005	Wegener and Grochtmann, 1998 Mueller, 1998 Puschner, 1998 Wegener, 1999 Groß 2000,2001 and 2003 Tilli, 2006	Di Penta, 2007 Garoussi,2006 Garoussi,2008 Garoussi,2010
SA			Tracey,1998
Customized Algorithm		Pohlheim,1999	

IAdapter



IAdapter

Apache JMeter (2.13 r1665067)

Test Plan
nauberphkwork@WorkLoad Thr
WorkBench

nauberphkwork@WorkLoad Thread Group

Name: nauberphkwork@WorkLoad Thread Group

Comments:

[Help on this plugin](#) v1.0

Action to be taken after a Sampler error

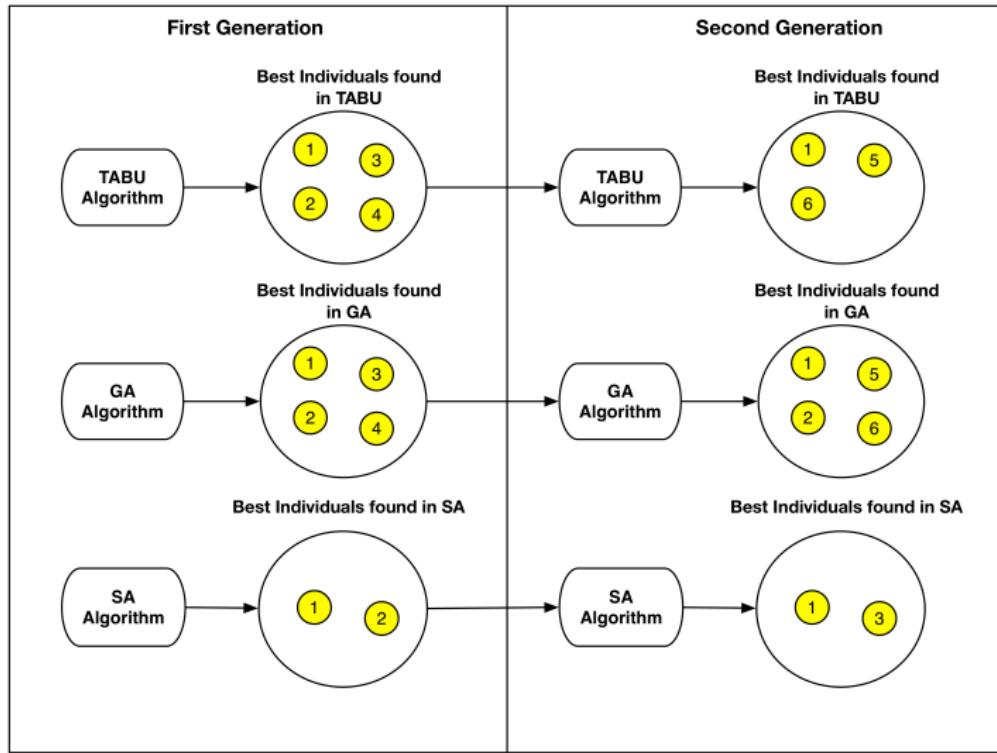
Continue Start Next Thread Loop Stop Thread Stop Test Stop Test Now

Main WorkLoad Graph Agents Parameters Log FIT Population

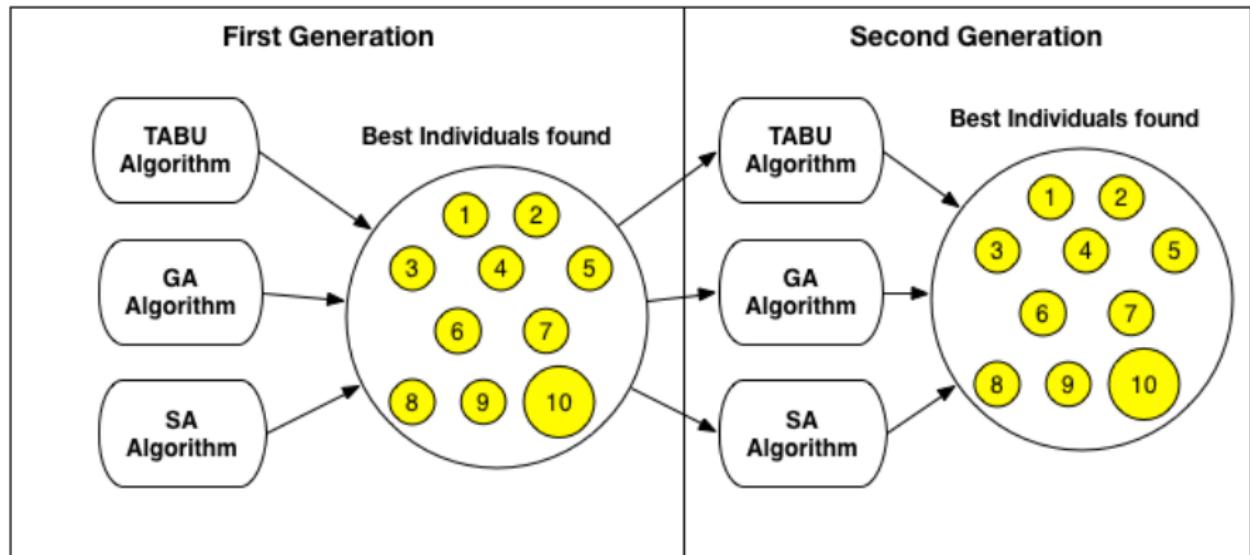


The IAdapter logo consists of five colored circles (red, blue, green, yellow, purple) arranged in a cluster, with the word "IAdapter" written in a bold, black, sans-serif font next to them.

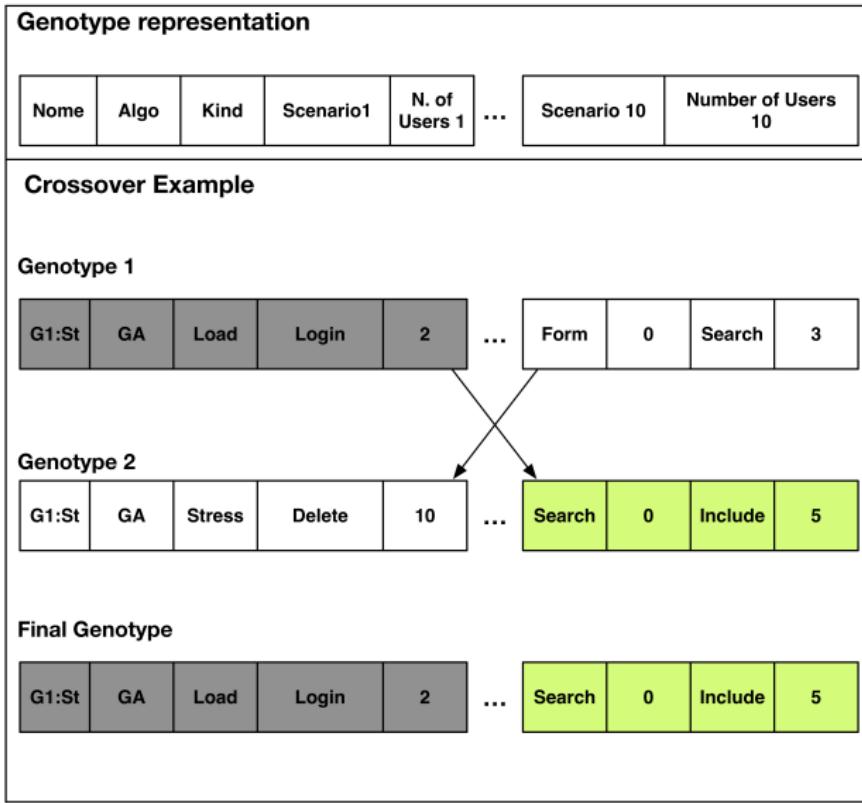
IAdapter-Independent approach



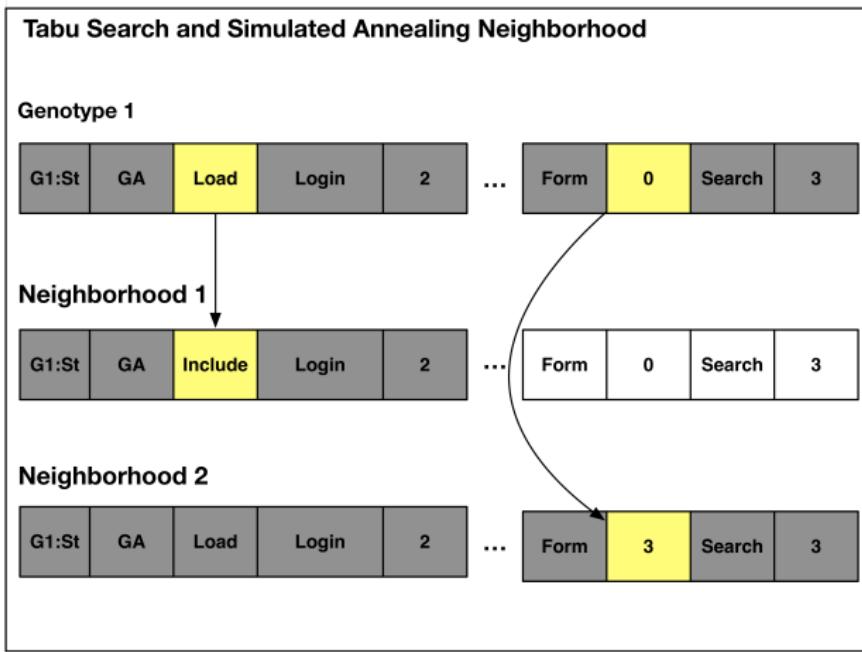
IAdapter-Collaborative Approach



IAdapter-Genotype Representation



IAdapter-Tabu Search



IAdapter-Fitness Function

$$\begin{aligned} fit = & 90\text{percentileweighth} * 90\text{percentiletime} \\ & + 80\text{percentileweighth} * 80\text{percentiletime} \\ & + 70\text{percentileweighth} * 70\text{percentiletime} + \\ & maxResponseWeighth * maxResponseTime + \\ & numberOfUsersWeighth * numberOfUsers - penalty \end{aligned}$$

Experiments

- The first experiment has implemented 27 generations;
- The second experiment has performed 6 generations;
- 300 executions by generation (100 times for each algorithm), generating 300 new individuals;
- The experiments had used a initial population of 100 individuals;
- The Genetic Algorithm used the top 10 individuals from each generation to the crossover operation;
- The Tabu List has been configured with the size of 10 individuals and expire every 2 generations;
- The mutation operation was applied to 10% of the population on each generation.

Listing 1: SimulateConcurrentAccess class

```
1 public class SimulateConcurrentAccess {  
2     @Test  
3     public void test() {  
4         synchronized (StaticClass.class) {  
5             for (int i = 0; i <= 1000; i++) {  
6                 StaticClass.x += i;  
7             }  
8             StaticClass.x = 0;  
9         }  
10    }
```

First Experiment I

Tabela : Fitness function maximum value by algorithm (90% percentile)

GEN	HM	TS	GA	SA
1	11238	11238	11238	11238
2	11804	11596	11801	10677
3	11787	8932	8411	10869
4	11723	9753	9611	10760
5	8164	9780	10738	4794
6	11802	9781	11086	6120

First Experiment I

Tabela : Fitness function maximum value by algorithm (90% percentile)

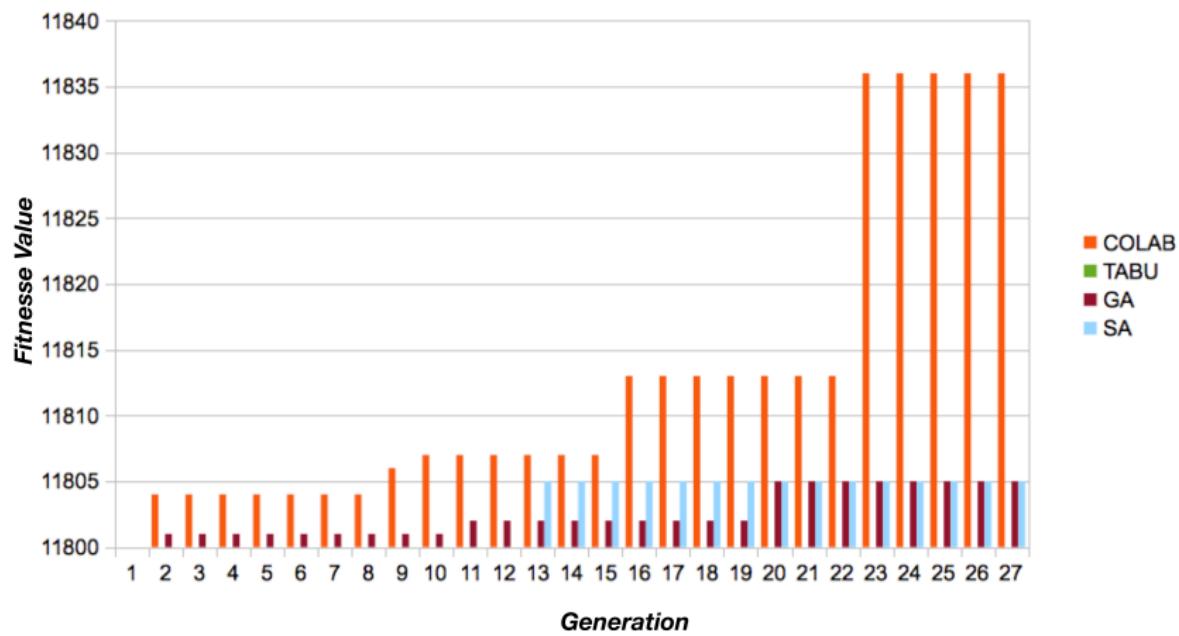
GEN	HM	TS	GA	SA
7	9985	5782	11272	11798
8	11803	11749	10084	11309
9	11806	7284	11633	10766
10	11807	9386	11717	4557
11	11802	9653	11802	11151
12	11807	10594	11793	9434
13	11802	10848	10382	11805
14	11801	11551	7219	10237
15	11807	1701	7189	9338
16	11813	6203	11758	5321
17	11805	10720	10805	11748

First Experiment I

Tabela : Fitness function maximum value by algorithm (90% percentile)

GEN	HM	TS	GA	SA
18	9600	6371	11698	7818
19	11733	8160	11648	11509
20	9589	9428	11805	4813
21	11800	9463	11798	10801
22	11805	11799	11804	6029
23	11836	11655	11800	3579
24	11805	11512	11803	5761
25	11804	11573	11802	9680
26	11800	11575	11403	9388
27	11805	10691	11745	9465

Second Experiment



Wilcoxon Test

- The Wilcoxon signed-rank test is a non-parametric statistical hypothesis test used when comparing two related samples, matched samples, or repeated measurements on a single sample to assess whether their population mean ranks differ.
- The Wilcoxon test applied in the first experiment showed a significant advantage of hybrid using metaheuristic approach.

Second Experiment

The screenshot shows a Moodle site administration interface. At the top, the Moodle logo is displayed with a graduation cap icon. The top right corner shows the message "You are logged in as Admin User (Logout)" and a language dropdown set to "English (en)".

The left sidebar, titled "Site Administration", contains the following menu items:

- Notifications
- Users
- Courses
- Location
- Language
- Modules
- Security
- Appearance
- Front Page
- Server
- Networking
- Reports
- Miscellaneous

Below the sidebar is a search bar with the placeholder "Search".

The main content area has a header "Available Courses" and a sub-header "No courses in this category". It includes a button "Add a new course".

To the right, there is a "Turn editing on" button and a "Getting started..." section with the following instructions:

- Login as **admin** with password **turnkey** and get moodling.
- Refer to the [TurnKey Moodle release notes](#)

At the bottom right is a "Calendar" section showing the month of September 2009. The calendar grid is as follows:

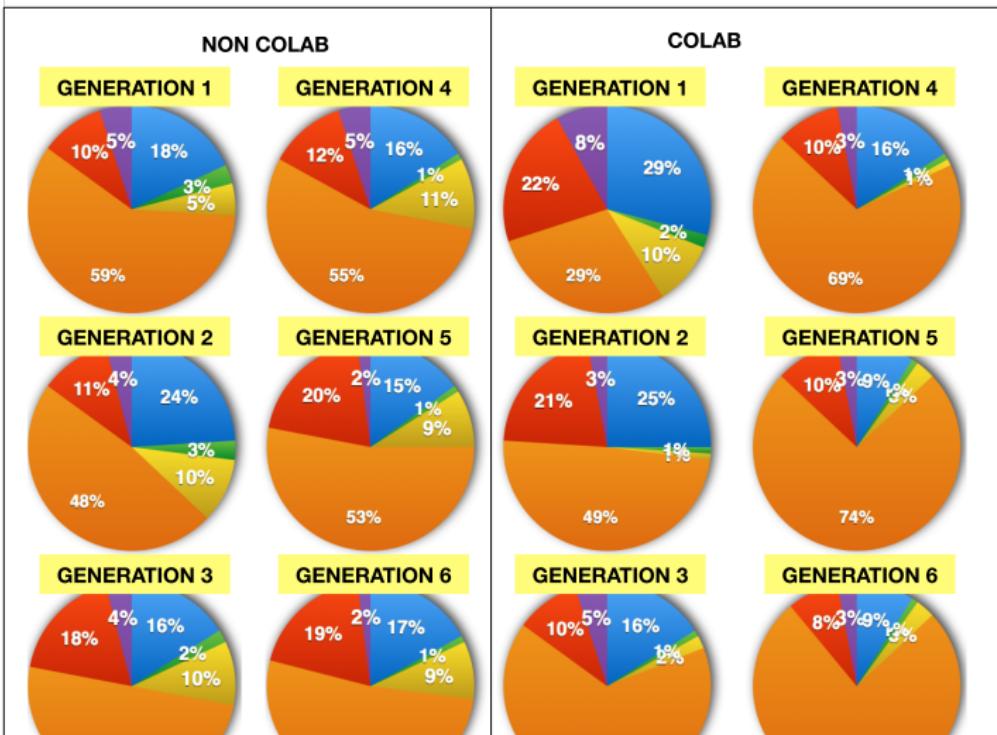
Sun	Mon	Tue	Wed	Thu	Fri	Sat
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30			

At the very bottom of the page, there is a footer bar with various navigation icons and the text "You are logged in as Admin User (Logout)".

- PostDeleteMessage- This scenario post and delete messages in the moodle application.
- MyHome- This scenario access the user's homepage of the application.
- Login- This scenario are responsible by the user authentication of the application.
- Notifications- This scenario enter in the notification page of each user.
- Start Page- Initial start page of the application.
- Badge- This scenario enter in the Badge page.

Second Experiment

BADGES POST NOTIFICATIONS MYHOME START PAGE LOGIN



Second Experiment I

Tabela : Results obtained from the second experiment

GEN	HM	TS	GA	SA
1	32242	32242	32242	32242
2	34599	32443	26290	35635
3	35800	34896	34584	34248
4	35782	34912	32689	25753
5	35611	31833	34631	8366
6	35362	35041	33397	9706

Second Experiment

Table 3: Example of individuals obtained in the second experiment

Ind	Fit	Scenario	N.Users	90per	80per	70per
1	35800	MyHome	31	30	29	10
		Badges	4			
2	35795	MyHome	30	30	29	10
		Notifications	2			
		Badges	2			
3	35782	MyHome	32	30	29	10
		Badges	3			
4	35773	MyHome	22	30	29	10
		Notifications	6			
		Badges	9			
5	35771	MyHome	28	30	29	9
		Badges	6			
6	35683	MyHome	27	30	29	8
		Badges	10			

Schedule

Ant Colony Algorithm development in the solution until
01.01.2016

Application of new experiments until
01.03.2016

Article published in magazines and periodicals until
01.07.2016

Distribution of the IAdapter in the JMeter Community until
01.07.2016

Qualification until
01.08.2016

Evaluation of iadapter
by users of jmeter
community unit
01.12.2016

Thesis presented until
01.07.2017

Conclusion

- This research presented a approach of use Hybrid Metaheuristic in load, performance and stress testing
- Two experiments were performed to validate the solution. The first experiment has been applied in an emulated component and the second experiment has been applied in an installed Moodle application.
- The collaborative approach has obtained better fit values in both experiments.

Bibliografia I



(2009).

A systematic review of search-based testing for non-functional system properties.

Information and Software Technology, 51(6):957–976.



Di Lucca, G. a. and Fasolino, A. R. (2006).

Testing Web-based applications: The state of the art and future trends.

Information and Software Technology, 48:1172–1186.



Molyneaux, I. (2009).

The Art of Application Performance Testing.

"O'Reilly Media, Inc." .



Sandler, C., Badgett, T., and Thomas, T. (2004).

The Art of Software Testing.

page 200.

Bibliografia II

-  Sullivan, M. O., Vössner, S., Wegener, J., and Ag, D.-b.
Testing Temporal Correctness of Real-Time Systems — A New
Approach Using Genetic Algorithms and Cluster Analysis —.
pages 1–20.