Paper 12

©Civil-Comp Press, 2011
Proceedings of the Thirteenth International Conference on
Civil, Structural and Environmental Engineering Computing,
B.H.V. Topping and Y. Tsompanakis, (Editors),
Civil-Comp Press, Stirlingshire, Scotland

Improving Engineer Efficiency in the Design of Railway Electrical Systems by using a High Performance Computing Platform

E. Arias, T. Rojo, D. Cazorla and F. Cuartero
Albacete Research Institute of Informatics
Albacete, Spain

Abstract

In order to make an optimal design of the interaction pantograph/catenary in railways it is necessary to ensure the correct placement of the line from where the pantograph obtains energy. The loss of contact between the pantograph head and the catenary produces a loss of energy in the train and, consequently, its speed decreases. Nowadays, the Spanish railway company uses a software tool called CALPE in the development of catenary electrical systems. However, due to the fact that this application has almost one hundred parameters, each one varying in a range, the amount of simulations that need to be carried out to cover a wide range of configurations is enormous, and CALPE tool remains itself insufficient. Thus, a High Throughput Computing (HTC) platform could be useful to achieve this purpose. In this paper, a HTC platform based on BOINC is presented to solve the static problem on the pantograph/catenary interaction. By using this platform, an efficient management of the computational resources of the organization is achieved which involves a clear benefit for this organization.

Keywords: high speed railways, pantograph-catenary interaction, resource management, high throughput computing, BOINC.

1 Introduction

In the last years, railway transport has evolved by adapting itself to new user requirements, which demand for greater comfort and safety. This, however, is bound up with a decrease in journey time, which implies an increase in the speed of the train.

The high-speed trains have a limit on their speed up to 350 Km/hour. In order to reach these speeds, it is important to provide constant energy to the train. For this reason, it is necessary to carry out an optimal design of the interaction pantograph/catenary, ensuring suitable placement of the line, with the contact wire parallel to the ground, so that the pantograph can obtain the energy from the line. The loss of contact between the pantograph head and the catenary produces in the train a loss of energy and, consequently, a decrease in the speed. Moreover, there is another problem because we have to guarantee the permanent contact between the pantograph and the catenary but without increasing the noise and wear.

In [2, 3, 12] the authors presented a High Performance implementation that was carried out on various algorithms to solve the equilibrium equation of the system in a catenary model, using spans with equal and different numbers of droppers and equal and different positions. The implementations have been performed both for the continued and stitched structure of the catenary. By means of these algorithms, it is possible to solve the static problem [14, 7, 12] of the system position that is determined when vertical force is exercised on the contact wire. If the stiffness matrix K and the vector of independent terms r of the system is known, it is possible to solve the linear system for the position of nodes y of the wire, and thus obtain the balance position. All these algorithms have been used as part of the tool CALPE 6.0 [8] that, at present, is used by the Spanish railway company as an important tool in the development of catenary electrical systems. Also, it is possible to solve the dynamic pantograph/catenary interaction as it is discribed by the authors in [4]

However, the amount of parameters to be consider in order to obtain an optimal installation of the pantograph/catenary system is considerable (almost 100). If, in addition, it is considered that the installation of a set of spans covers 1200 m approximately, and the considerable amount of kilometers in railways in Spain, it is easy to imagine the great amount of simulations that need to be carried out considering the variability of the parameters. Even thought, taking into account the expertise of the engineers in order to reduce the number of tests (in this case the engineers belong to ADIF [1], the company in charge of the railway infrastructures in Spain), this number is still very high.

One good feature which benefits this particular application is that the execution time is short, between 2 or 3 minutes. Thus, by means of CALPE, the engineer design a particular electrical installation of catenary using a try and error way. This is because no analytic formula may be obtained to calculate the results, and it is necessary the use of simulations made by CALPE. But if one simulation designed by the engineer is not working, then, it is necessary to modify some parameters, and try a new simulation. And so on, until a good simulation is obtained, with the criteria of the engineer. This is a tedious work, and it is possible to do several simulations at the same time, by using a distributed platform. Moreover, the short execution time of one simulation allows that the target platform carrying out the test does not need to have special hardware requirement in terms of main memory and speed of processor. Also, these low computational requirements make this application ideal to use not only on a dedicated platform, but also on a non-dedicated one, taking benefits of the idle processor cycles when the staff of the company is not working.

This means that we can use not only High-Throughput Computing batch systems on dedicated cluster (such as Condor [18], PBS[15], and Sun Grid Engine[17]) but also distributed computing on client computers using the BOINC platform [6], which allows us to use the computer power of client computers when these are idle. Taking into account the previous considerations of this application, became natural the use of a High Throughput Computing Platform (HTC) to carry out hundreds of simulation without human intervention (only when the test are generated).

High-Throughput Computing depends on the availability of computational cycles. Some applications require weeks and even months of computational task in order to obtain a result. Thus, the idea is not to obtain a high speed on computation during a short period of time, but to obtain and keep the necessary resources along the time in order to make some computation. The batch systems on a cluster allow to accomplish the previous assumptions. Then, one or more nodes in a cluster can be assigned during some period of time ensuring the exclusive use of the nodes by the job. Among the more popular batch systems, we can find Condor [18], PBS[15], and Sun Grid Engine[17].

Between all the previous software, BOINC [6] has been chosen because it accomplishes the conditions previously outlined for our application and its implantation at the enterprizes with guarantees of success.

As far as we know, there are a few software packages which compute the static problem of the pantograph/catenary interaction [5, 13]. However, none of them use a HTC platform to optimized the use of the computational resources available at the enterprizes. The aim of this paper is to show how the BOINC platform has been used in order to solve the pantograph/catenary interaction problem using distributed computing on non dedicated computers. This new approach allows us to change the engineers methodology, changing from a trial and error methodology to a search for good/best solution methodology.

The paper is structured as follows. In section 2 some of the features of the static stiffness problem in a catenary model are described. After that, a brief overview of the BOINC platform is introduced in section 3. In Section 4 Calp@home, the BOINC application, is presented. Finally, the conclusions and future work are commented in Section 5.

2 Description of the problem

There is a well established sequence of steps that we need to follow to model a simulation, and doing the corresponding experiments. This sequence is the following:

1. Formulation of the problem
2. Generating the model
3. Obtention of the corresponding data
4. Implementation of the model in the computer

5. Making the proofs
6. Validation of the model
7. Design of the simulation experiments
8. Running the experiments
9. Analysis of results
10. Final Report

To formulate the problem, the static stiffness problem gives us an estimation on the ideal assembly parameters of the catenary, and will also serve as an introduction to the dynamic problem [11], where the evolution of the pantograph is studied with the movement of the train. Then, we can generate the catenary model which is shown in Figure 1.

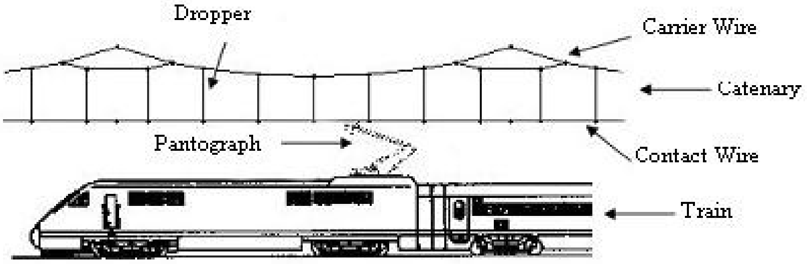


Figure 1: Catenary model with continued structure using spans with different numbers of droppers and positions.

In the static problem the static position of the system is determined when we apply a vertical force over the contact wire. So, to obtain the data, it is necessary to configure the stiffness matrix K and the independent term R of the system and then solving the linear system for the position of the nodes Y of the wire, and can thus obtain the balance position.

$$K * y = r \quad (1)$$

After that, it is needed to check if all droppers work in a traction mode. In an affirmative case the problem is solved, but in the negative case, the stiffness matrix and the independent term must be reconfigured in order to eliminate the terms of the droppers that do not work, obtaining the new position and repeating this problem until all droppers work in a traction mode.

Once the equation that determines the position of every node forming the structure of the catenary has been solved, it is necessary to tackle another study that will produce an iterative process. By taking a point inside the first span we exercise a vertical force on the contact wire, so that we must solve the equilibrium equation again, to obtain the new position of every node inside the structure of the catenary. This process is iterated for several points inside every span. Once the study has been performed inside the first span, the same process has to be repeated for each of the remaining spans.

This process is explained in Figure 2, and it is actually implemented in the CALPE tool. This tool has been validated by Adif engineers, using several techniques and experiments testing the results of the simulations against real measures.

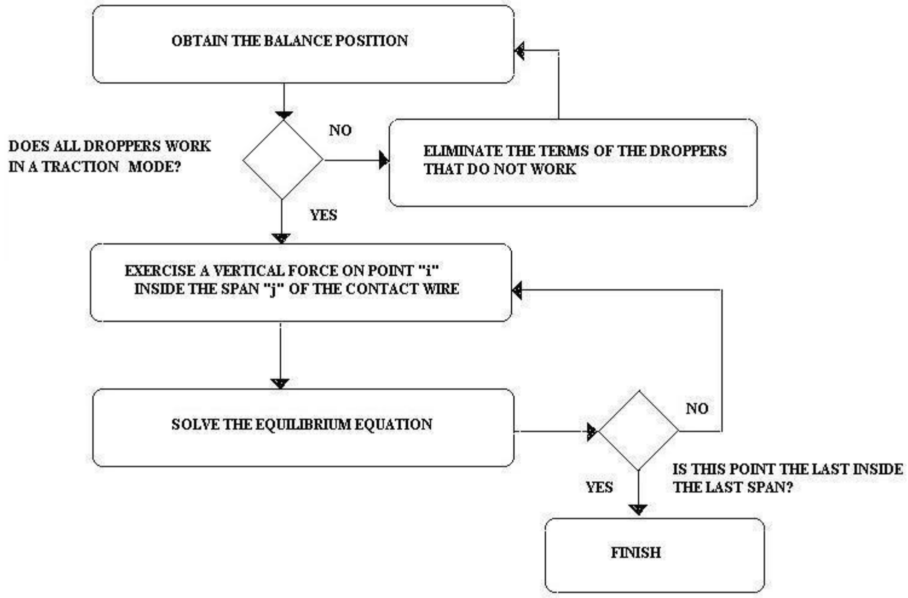

OBTAIN THE BALANCE POSITION
NO
DOES ALL DROPPERS WORK IN A TRACTION MODE?
ELIMINATE THE TERMS OF THE DROPPERS THAT DO NOT WORK
YES
EXERCISE A VERTICAL FORCE ON POINT "i" INSIDE THE SPAN "j" OF THE CONTACT WIRE
SOLVE THE EQUILIBRIUM EQUATION
NO
IS THIS POINT THE LAST INSIDE THE LAST SPAN?
YES
FINISH

Figure 2: Scheme of the static problem.

Once the tool is useful in the real life, we formulate a new approach, which seems to be interesting to the engineers using CALPE. When they are designing a new installation, they follow a test and error methodology, designing a model in the first stage, simulating it, and validating the results so obtained. If they are not interesting, the model is changed, in an arbitrary way, following the engineer criterium and his know how, this process is repeated until the obtained results are satisfying the engineer. Of course, this process is tedious, and it is a wasted of precious time. This, it is interesting to select some parameters in the input data, varying them in a determined range, and select the output data, in order to obtain the interesting results. In this way, varying the parameters, we obtain a set of different models, which can be run in parallel in a distributed system. Taking into a account that each possible simulation takes about 2 minutes, and that no communication is necessary between different simulations, because they are independent of each other, as much as nodes we have in the system, as quicker will be the full simulation; and the main node only needs to distribute the different simulations among the rest of the nodes, and to obtain the corresponding results, if they are satisfying the output criteria. When all the simulations are finished, the engineer has a set of interesting models, and he only needs to select the best using his know how, and other not automatized criteria. Thus, the improvement of the efficiency of the task of the engineer is considerable, from the analysis of, perhaps, about a hundred of possibilities in several days or even weeks, to the analysis of several hundred of thousands in a single day.

3 BOINC: a software platform for distributed computing

The next step is to implement the model in the computer. We have selected a BOINC platform, where our application, called CALP@HOME, will be implemented. BOINC (Berkeley Open Infrastructure for Network Computing) [9, 6], developed at the U. C. Berkeley Spaces Sciences Laboratory, is an open software platform for distributed computing that uses volunteered computing and desktop grid computing, i.e., it uses the idle time on volunteers computer to compute different applications, such us studying climate change or searching extraterrestrial intelligence. BOINC consists of a server system and a client software that communicate with each other to distribute, process, and return work units.

Computations are run on volunteers computers, then uploaded from the user computer to the server project database, and finally the results are analyzed and validated by the server BOINC software. The validation process involves running all tasks on multiple contributor computers and comparing the results.

The client side consists of the BOINC core, that communicates with external servers to get and report work, and the applications that are the programs that do scientific computing. Several of them may run at the same time on the volunteer computer.

The main features of the BOINC system are the following:

- Flexible application framework. BOINC approach can be easily used with a wide range of applications. Little or no modification needs to be done on the source code, mainly at the beginning and at the end of the code.

- Security. Digital signatures based on public-key encryption are used in order to avoid distribution of fake applications (that may contain viruses) that supplant the right ones.

- Multiple servers and fault-tolerance. A project may use several servers so, in case one of the servers is temporarily offline, the client software communicates with an alternative server.

- Source code availability. BOINC is distributed under the Lesser GNU Public License.

- Share resources among several projects. Client software allows volunteers to decide which projects they enrol and which percentage of his/her resources are dedicated to each project.

- Rewards participants. BOINC provides a credit accounting system that reflects usage of several resources. Most active users may obtain different rewards, depending on the project.

There are projects using the BOINC philosophy in several fields of science such us biology and medicine (Rosetta@home, proteins@home), climate modelling (Climateprediction.net), physics and astronomy (LHC@home, SETI@home, Einstein@Home), mathematics (ABC@Home).

The most popular BOINC project is SETI@home [10, 16], also developed at the U. C. Berkeley Spaces Sciences Laboratory. The initial software platform, now referred to as "SETI@home Classic", was only capable of running SETI@home; recently it was replaced by a BOINC version, allowing users to contribute to other distributed computing projects at the same time.

SETI@home was, in fact, the origin of the BOINC platform because the success of SETI@home made it clear that distributed computing could be used for many other computing-intensive scientific projects.

4 CALP@HOME: A tool based on BOINC for Pantograph/Catenary Interaction Studies

In order to make a good design of the interaction pantograph/catenary, the engineer needs to give value to a set of parameters and, afterwards, he/she has to use several algorithms to solve the equilibrium equation of the system in the catenary model. Once results are obtained, a post-analysis is necessary in order to know whether or not the solution fits to the specifications, mainly to minimize the loss of contact between the pantograph and the catenary. If the solution is not good enough, the value of one or several parameters must be changed and then the process starts again.

This approach to the problem uses a trial and error methodology, and is based on engineers expertise. An experienced engineer may drive the analysis in order to reach an acceptable solution after several computations.

Nevertheless, optimal solutions are difficult to find because a great amount of parameters are involved at each simulation, and engineers focus on a few of them. In order to give a full view of the problem, a non-exhaustive list of parameters that need to be considered is shown in Fig 3

In order to understand the need of a HTC distributed platform, only considering the subset of parameters in Table 1 and considering different values for these parameters, the number of simulations that will be carried out is over 25 million.

From our point of view, it is quite clear the need of a change in the methodology.

- Type of catenary systems: simple continued, simple alternative, stitched continued and stitched alternative.
- Number of spans.
- Number of droppers.
- Length of the section.
- Type of droppers.
- Distance between the first dropper and the support.
- Distance between droppers.
- Weight of the spanner of the carrier.
- Weight of the spanner of the contact wire.
- Section of droppers.
- Specific weight of the material of the droppers.
- 8 parameters with information about the carrier.
- 6 parameters with information about the contact wire.
- Height of support of the carrier.
- Maximum length in elements of the carrier.
- Maximum length in elements of the contact wires.
- Information about the arm (5 parameters).
- Number of pantographs.
- First position of pantograph '#p'
- Force to consider separated the wire from the pantograph.
- Type of pantograph.
- Length of contact area of the pantograph.
- Distance between platens.
- Weight mass.
- Stiffness associate.
- Dimming.
- Upward statics force.
- Friction force.

Figure 3: Non-exhaustive list of parameters

Instead of a trial end error methodology where the engineer modifies by hand several parameters in order to get a good solution, a new approach can be considered where the engineer defines a set of parameters that can get different values over a range and then the distributed platform runs every simulation and returns the best result.

This methodology has three main steps:

1. From the engineer specification, a set of input files are automatically generated (work units in BOINC terminology).

2. Every input file is sent to a BOINC client, which runs the simulation and returns the result to the server.

3. A post-processing, taking into account critical parameters in the output files, allows the engineer to obtain directly the best solution.

This new methodology has several advantages that can be summarized as: design time is shortened, solutions are better than before, and an efficient use of the computational resources available at the organization is achieved, all of these without additional cost.

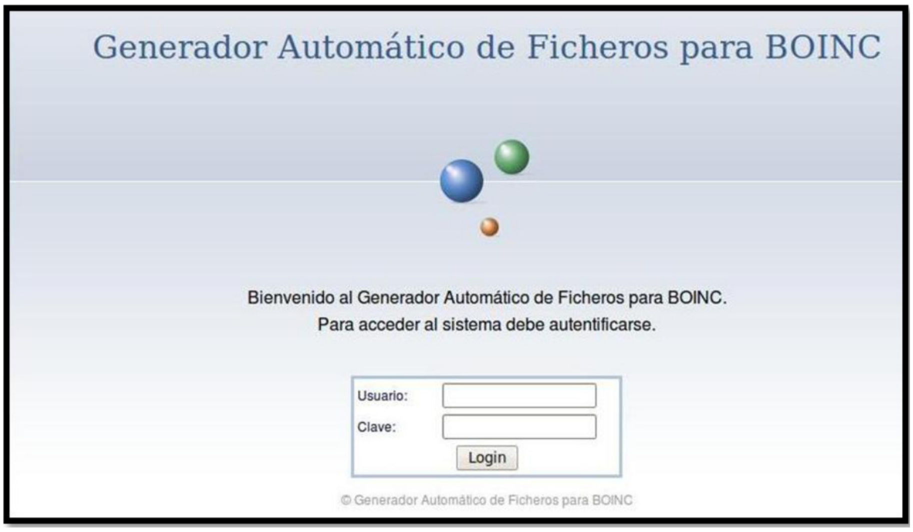
To implement this HTC tool several software has been used: BOINC, Apache server, mysql, shell scripts, PHP, etc.

Parameter	Values	Variations
Type of catenary system		4 possibilities
Number of spans	$[12-20]$	10 possibilities
Number of pantographs	$[1-4]$	4 possibilities
Type of pantographs		4 possibilities
Velocity of train	$[250Km/h-350Km/h]$	10 possibilities
Span length	$[40m-60m]$	10 possibilities
Contact wire height	$[5m-6m]$	10 possibilities
Number of contact wires		2 possibilities
Number of droppers	[10-20]	10 possibilities

Table 1: Subset of parameters involved at the pantograph/catenary interaction

BOINC allows implement the HTC tool, and it is supported by Apache server.

In order to carry out the authentication stage to the Calp@home tool mysql software has been used (see Figure 4).



Generador Automático de Ficheros para BOINC

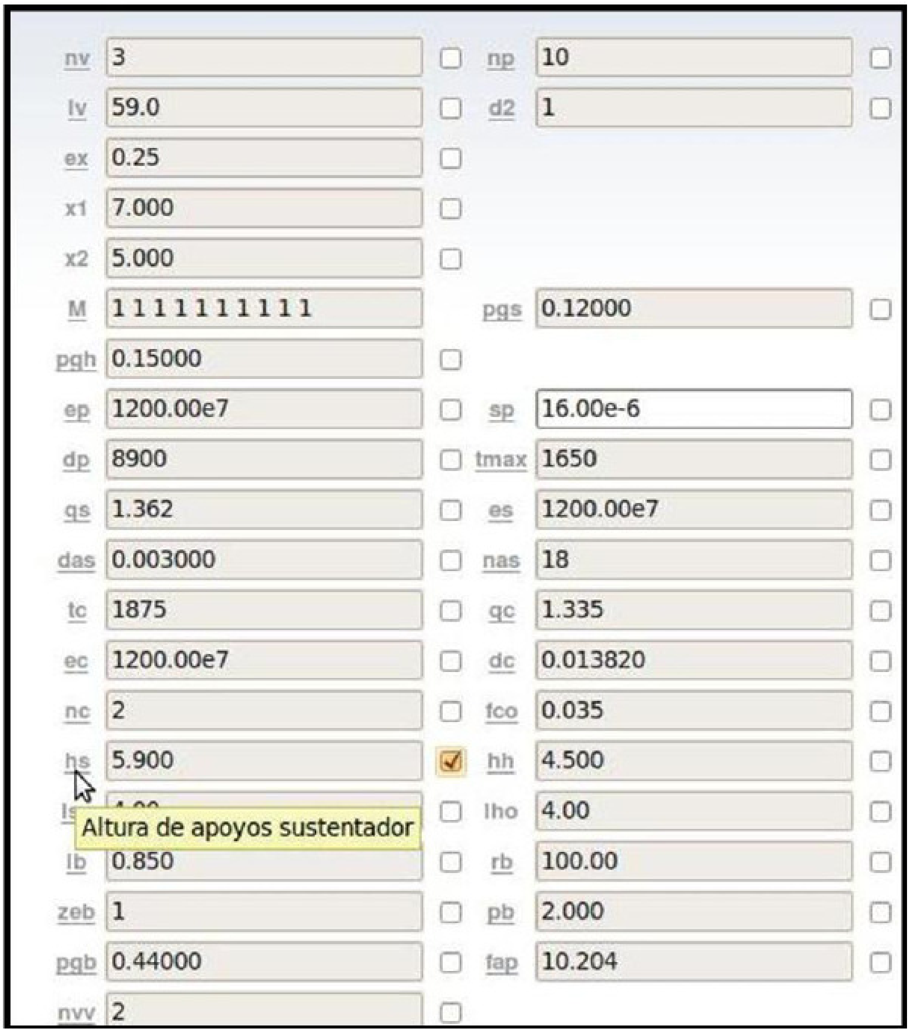
Bienvenido al Generador Automático de Ficheros para BOINC.
Para acceder al sistema debe autentificarse.

Figure 4: User authentication.

Once the user has been authenticated, the user can load a file from he/she directory (see Figure 5). This file contains configuration parameters for the pantograph/catenary interaction computations.

Now, the user can choose one of more parameters and introduce initial value, final value and increment of the parameter (see Figure 6).

When the parameters have been appropriately modified, the user launch the CALPE application (really, the user launch several configuration files), and when the computations have been finished a shell script analyse the resulting file and, according to a predifined optimization function, an e-mail is sent to the user with the name of the



nv	3	□	np	10	□
lv	59.0	□	d2	1	□
ex	0.25	□			
x1	7.000	□			
x2	5.000	□			
M	1 1 1 1 1 1 1 1 1 1		pgs	0.12000	□
pgh	0.15000	□			
ep	1200.00e7	□	sp	16.00e-6	□
dp	8900	□	tmax	1650	□
qs	1.362	□	es	1200.00e7	□
das	0.003000	□	nas	18	□
tc	1875	□	qc	1.335	□
ec	1200.00e7	□	dc	0.013820	□
nc	2	□	fco	0.035	□
hs	5.900	☑	hh	4.500	□
ls	4.00	□	lho	4.00	□
lb	0.850	□	rb	100.00	□
zeb	1	□	pb	2.000	□
pgb	0.44000	□	fap	10.204	□
nvv	2	□			

Altura de apoyos sustentador

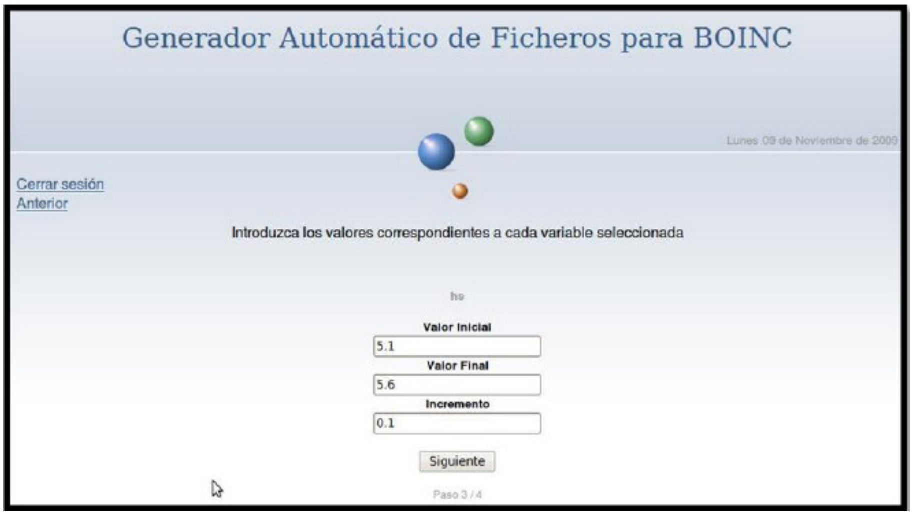
Figure 5: Example of file configuration load by the user.

configuration file with the optimal configuration of the pantograph/catenary structure (see Figure 7).

5 Conclusions

In this paper a High Throughput Computing (HTC) Platform has been introduced for solving a problem of civil engineering, in particular, a simulation of the pantograph/catenary interaction in order to obtain the optimal parameters for the installation.

This HTC platform has been implemented using BOINC software. This software allows an efficient use of the computational resources due to the fact that the resources are used even during the idle periods, running automatically an application with the



Generador Automático de Ficheros para BOINC

Lunes 09 de Noviembre de 2009

Introduzca los valores correspondientes a cada variable seleccionada

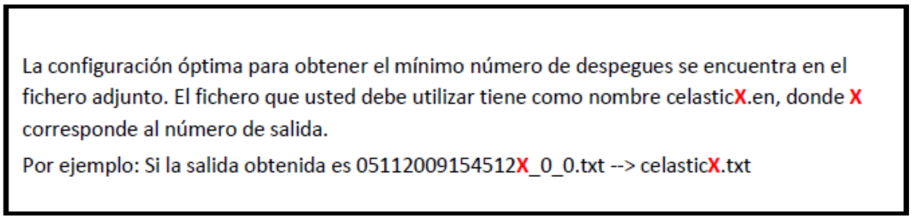
hs

Valor Inicial: 5.1

Valor Final: 5.6

Incremento: 0.1

Figure 6: A screen to modify the initial, final and increment of the parameters.



La configuración óptima para obtener el mínimo número de despegues se encuentra en el fichero adjunto. El fichero que usted debe utilizar tiene como nombre celasticX.en, donde X corresponde al número de salida.

Por ejemplo: Si la salida obtenida es 05112009154512X_0_0.txt --> celasticX.txt

Figure 7: E-mail informing about the best configuration file.

corresponding input data. By proceeding in this way a better management resources present at the organization is achieved without extra economical cost.

Indeed, the new methodology introduced by the use of the HTC platform permits the engineers to produce a huge number of test in order to decide empirically the best parameters for the installation of the pantograph/catenary system. As a result, the time spent on the design stage is reduced and better installations are obtained, improving considerable the efficiency of the engineer task.

Acknowledgements

We will thank Jesús Montesinos, engineer of ADIF, for its technical support and good knowledge of the real problem beyond the simulation world. Also, we will thank CALPE team by its invaluable support during the project.

References

[1] Administrador de Infraestructuras Ferroviarias, http://www.adif.es/.

[2] A. Alberto, E. Arias, D. Cebrian, T. Rojo, F. Cuartero, J. Benet, An Efficient Method to Solve the Static Stiffness Problem on a catenary with Equal Spans, The 14th European Conference on Mathematics for Industry, 2006.

[3] A. Alberto, E. Arias, D. Cebrian, T. Rojo, F. Cuartero, J. Benet, A High Performance Algorithm to solve the Static Stiffness Problem of a Catenary, in B.H.V. Topping, G. Montero, R. Montenegro, (Editors), Proceedings of the Eighth International Conference on Computational Structures Technology, Civil-Comp Press, Stirlingshire, UK, Paper 174, 2006. doi:10.4203/ccp.83.174

[4] A. Alberto, J. Benet, E. Arias, D. Cebria, T. Rojo, F. Cuartero, A high performance tool for the simulation of the dynamic pantograph-catenary interaction, Mathematics and Computers in Simulation, 79, 652-667, 2008.

[5] A. Balestrino, O. Bruno, A. Landi, L. Sani, PANDA: a friendly CAD tool for pantograph design and testing, Seventh International Conference on Computer in Railways, 2000

[6] Berkeley Open Infrastructure for Network Computing, http://boinc.berkeley.edu/.

[7] B. Simeon, M. Arnold, Coupling DAEs and PDEs for Simulating the interaction of pantograph and catenary, Mathematical and Computer Modelling of Dynamical Systems, 6, 2, 129-144, 2000.

[8] CALPE Project, http://www.dsi.uclm.es/personal/AntonioBueno/retics/retics.

[9] D.P. Anderson, BOINC: A System for Public-Resource Computing and Storage, 5th IEEE/ACM International Workshop on Grid Computing, 2004.

[10] D.P. Anderson, J. Cobb, E. Korpela, M. Lebofsky, D. Werthimer, SETI@home: An experiment in public-resource computing, Communications of the ACM, 45, 11, 55-61, 2002.

[11] E. Arias, J. Benet, P. Brui, F. Cuartero, T. Rojo, Solution of the Dynamical Problem of Pantograph-Catenary Interaction: A High Performance Computing Approach, in B.H.V. Topping, C.A. Mota Soares, (Editors), Proceedings of the Seventh International Conference on Computational Structures Technology, Civil-Comp Press, Stirlingshire, UK, Paper 230, 2004. doi:10.4203/ccp.79.230

[12] E. Arias, A. Alberto, J. Montesinos, T. Rojo, F. Cuartero, J. Benet, A mathematical model of the static pantograph/catenary interaction, Int. J. Comput. Math., 86, 2, 333-340, 2009.

[13] L. Drugge, T. Larsson, A. Stensson, Modelling and Simulation of Catenary-Pantograph Interaction. Vehicle System Dynamics, 33, 490-501, 2000.

[14] Poetsch G., J. Evans, R. Meisinger, W. Kortum, M. Baldauf, A. Veitl, J. Wallaschek, Pantograph/catenary dynamics and control, Vehicle System Dynamics, 28, 159-195, 1997.

[15] Portable Batch System, http://www.openpbs.org/main.html.

[16] SETI@home, http://setiathome.berkeley.edu/

[17] Sun Grid Engine, http://www.sun.com/software/gridware/.

[18] The Condor Project, http://www.cs.wisc.edu/condor/.