Optil.io: Cloud Based Platform for Solving Optimization Problems Using Crowdsourcing Approach

Szymon Wasik
Poznan University of Technology
Poznan, Poland
szymon.wasik@cs.put.poznan.pl

Maciej Antczak
Poznan University of Technology
Poznan, Poland
maciej.antczak@cs.put.poznan.pl

Jan Badura
Artur Laskowski
Tomasz Sternal
Poznan University of Technology
Poznan, Poland
jan.badura@student.put.poznan.pl
artur.laskowski@student.put.poznan.pl
tomasz.sternal@student.put.poznan.pl

CSCW '16 Companion, February 27 - March 02, 2016, San Francisco, CA, USA
ACM 978-1-4503-3950-6/16/02.
http://dx.doi.org/10.1145/2818052.2869098

Abstract

UPDATED—January 2, 2016. The main objective of the presented research is to design a platform for continuous evaluation of optimization algorithms using crowdsourcing technique. The resulting platform, called Optil.io, runs in a cloud using platform as a service model and allows researchers from all over the world to collaboratively solve computational problems. This is the approach that has been already proved to be very successful for data mining problems by web services such as Kaggle. During our project we adapted this concept for solving computational problems that require implementation of software. To achieve this we designed the on-line judge system that receives algorithmic solutions in a form of source code from the crowd of programmers, compiles it, executes in a homogeneous run-time environment and objectively evaluates using the set of test cases. It was verified during internal experiments at the Poznan University of Technology and it is now ready to be presented to wider audience.

Author Keywords

crowdsourcing; cloud computing; on-line judge; optimization; algorithms; operational research; platform as a service;

Figure 1: Optil.io platform's logotype.

ACM Classification Keywords

H.5.3 [Group and Organization Interfaces]: Computer-supported cooperative work; G.2.1 [Combinatorics]: Combinatorial algorithms

Introduction

The last few years, since 2006, when Amazon has launched a set of services called Amazon Web Services that allowed for a remote management of the computing resources, have been a period of rapid development of so-called cloud computing technologies. Today, this term is used to describe a variety of services, which allow for processing and sharing of information using the remote servers through the Internet [1]. One of the interesting applications of cloud computing are services that allow for automatic assesment of submitted software, so-called on-line judges [4]. These are sites that allow users to submit the source code of the algorithm, which is next compiled on a remote server, executed and the quality of proposed algorithm is assessed on the basis of a predefined set of problem specific test cases. This approach allows for an objective evaluation of algorithms that solve some specific computational problem in a homogeneous run-time environment. When using on-line judge, the author of the algorithm cannot influence the process of evaluation by modifying the run-time environment or altering the set of test cases for which the program will be assessed. The evaluation results obtained in such a way can be considered reliable.

In the Internet there are already several open sourced, on-line judge systems. These are platforms such as Open Judge System [7], SIO2, or Online Judge Plugin for Moodle. However, they serve primarily to organize algorithmic competitions in the Internet [3], and are hard to adapt directly for other applications.

Website	*Comp.*	*Awards (USD)*	*Community*
Kaggle	47	1 309 000	20 000
TopCoder MM	15	40 000	300
CrowdAnalytix	18	43 700	4 000
InnoCentive	90	2 341 500	20 000

Table 1: Websites dedicated for solving problems using crowdsourcing approach. For each website the number of competitions performed (i.e. problems solved) in 2013 are presented together with the total budget spent on awards and estimated size of users community.

On the other hand several websites have been recently implemented that are following the idea of crowdsourcing, supplemented occasionally with the cloud computing, in order to engage the community in solving the scientific or industrial problems [2]. These services allow users for publishing of non-trivial problems in the form of competitions, that are being solved by the crowd and awarded for the best solutions. However, all existing systems have certain disadvantages and shortcomings. First of all, types of problems which can be evaluated are very limited and usually related to the data mining field only. Secondly, most of them are not integrated with the on-line judge system (except for the TopCoder [5] that in turn supports very limited number of programming languages), which significantly hinders the successful transfer of the solutions' source code to the industry and scientific community. Table 1. presents some of the existing websites together with the number of competitions performed and budget spent on awards in 2013.

Problem characteristics

Scheduling, classification of images, determination of the values of the parameters in a dynamic systems are just a few examples of problems for which new optimization algorithms are continuously developed. In order to evaluate these newly developed algorithms, for the particular problem that has been already defined, it is necessary to assess its quality by comparison with other existing solutions. The quality of alternative algorithms is typically derived from scientific publications, which are written by the authors of these algorithms. As a result, each algorithm is usually assessed using some specific measure, defined by the author of the algorithm thus evaluation results are presented using this specific measure only. Therefore, based on these measures it is not possible to build a reliable ranking of all algorithms solving the particular problem. Even solutions of problems, for which a standard set of test cases, called benchmark, is defined, as the traveling salesman problem [6], may be difficult to compare with each other, since researchers have freedom in choosing of benchmark's subset, which will be used to carry out the experiment, resource limitations, or quality measures that are used to make a comparison. Another problem is the unavailability of the algorithms source code described in scientific publications. Without this source code it is impossible to reliably verify the quality of algorithms designed by other researchers and apply them in practical problems derived from industry or science.

Optil.io platform

The solution to the problems described above proposed by us is the use of an on-line judge system, which allows for continuous evaluation of algorithms that were designed to solve optimization problems, by compiling them from sources, executing in coherent run-time environment, and evaluating according to predefined benchmarks. This approach solves all considered problems, however, it requires consideration of some new difficulties. First of all, how to ensure safe execution of the source code from an external source in the cloud. Secondly, how to define and set run-time environment resources limits and weights associated with them in the objective function, used as a measure, that should be taken into account during the evaluation of submitted algorithms (i.e. processing time and memory allocation). Finally, an important issue is to create a consistent, reliable ranking of algorithms by aggregating the results obtained on multiple benchmarks.

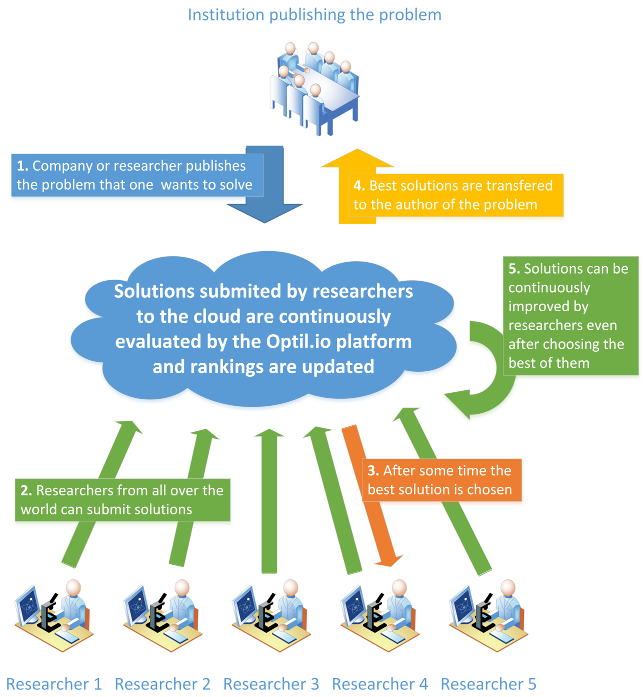
Institution publishing the problem
1. Company or researcher publishes the problem that one wants to solve
4. Best solutions are transfered to the author of the problem
Solutions submited by researchers to the cloud are continuously evaluated by the Optil.io platform and rankings are updated
5. Solutions can be continuously improved by researchers even after choosing the best of them
2. Researchers from all over the world can submit solutions
3. After some time the best solution is chosen
Researcher 1
Researcher 2
Researcher 3
Researcher 4
Researcher 5

Figure 2: The workflow of the Optil.io platform. Solutions can be subbmited in any programming language. Currently the platform supports C, C++, Python, Java, archives with CMake script and Linux static binaries but it can be easily extended.

We have solved the most of these difficulties by implementing an Optil.io platform which allows for publishing optimization problems together with problem-oriented testing benchmarks and methodology used to evaluate them and applies crowdsourcing as a solving technique for researchers all over the world. The architecture of the proposed platform is presented in Figure 2. Various organizations, academic or industrial, can submit the description of their own problems that need to be solved by a crowdsourced community. Researchers can then solve them, submit source code of the solution to the on-line judge system that executes them in the cloud and instantly evaluates. Each participant can observe how his solution is being compared to others and improve it further. It is possible to organize competitions following the deadlines from the organization that is interested to obtain the solution of the problem in a specific time frame or never-ending competitions to continuously improve solutions of scientific challenges. Problems can be submitted in almost any programming language. Then submitted solutions are carefully assessed taking into account resource limitations set for run-time environment, such as CPU time, virtual memory or disk space. This approach has been already implemented and tested by the group of 90 students at the Poznan University of Technology. Currently the platform is tested on a larger group of users.

Acknowledgements

All authors were supported by the Polish National Center for Research and Development under grant LIDER/004/103/L-5/13/NCBR/2014.

REFERENCES

1. Mariana Carroll, Paula Kotze, and Alta van der Merwe. 2012. Securing Virtual and Cloud Environments. In *Cloud Computing and Services Science*, Ivan Ivanov, Marten van Sinderen, and Boris Shishkov (Eds.). Springer New York, 73–90. DOI: http://dx.doi.org/10.1007/978-1-4614-2326-3_4
2. A. Goldbloom. 2010. Data Prediction Competitions – Far More than Just a Bit of Fun. In *2010 IEEE International Conference on Data Mining Workshops (ICDMW)*. 1385–1386. DOI: http://dx.doi.org/10.1109/ICDMW.2010.56
3. Vivek Khera, Owen Astrachan, and David Kotz. 1993. The Internet Programming Contest: A Report and Philosophy. *SIGCSE Bull.* 25, 1 (March 1993), 48–52. DOI:http://dx.doi.org/10.1145/169073.169105
4. Andy Kurnia, Andrew Lim, and Brenda Cheang. 2001. Online Judge. *Computers & Education* 36, 4 (2001), 299 – 315. DOI: http://dx.doi.org/10.1016/S0360-1315(01)00018-5
5. Karim R Lakhani, David A Garvin, and Eric Lonstein. 2010. Topcoder (a): Developing software through crowdsourcing. *Harvard Business School Case* 610 (2010), 032. http://hbr.org/product/TopCoder--A---Developing-/an/610032-PDF-ENG
6. Gerhard Reinelt. 1991. TSPLIB - A Traveling Salesman Problem Library. *ORSA Journal on Computing* 3, 4 (1991), 376–384. DOI: http://dx.doi.org/10.1287/ijoc.3.4.376
7. Shen Yunfu and Dong Xiaowen. 2011. Research on teaching reform of computer science and technology speciality in general education. In *2011 6th International Conference on Computer Science Education (ICCSE)*. 1463–1467. DOI: http://dx.doi.org/10.1109/ICCSE.2011.6028906