Research Article

Nikolay P. Khrapov*, Valery V. Rozen, Artem I. Samtsevich, Mikhail A. Posypkin, Vladimir A. Sukhomlin, and Artem R. Oganov

Using virtualization to protect the proprietary material science applications in volunteer computing

https://doi.org/10.1515/eng-2018-0009
Received Oct 02, 2017; accepted Dec 11, 2017

Abstract: USPEX is a world-leading software for computational material design. In essence, USPEX splits simulation into a large number of workunits that can be processed independently. This scheme ideally fits the desktop grid architecture. Workunit processing is done by a simulation package aimed at energy minimization. Many of such packages are proprietary and should be protected from unauthorized access when running on a volunteer PC. In this paper we present an original approach based on virtualization. In a nutshell, the proprietary code and input files are stored in an encrypted folder and run inside a virtual machine image that is also password protected. The paper describes this approach in detail and discusses its application in USPEX@home volunteer project.

Keywords: USPEX, VASP, BOINC, optimization, virtualization, structure prediction, computational materials design

1 Introduction

Due to the rapid progress in the computational power and optimization algorithms in the last 10-15 years, it became possible to predict new materials in-silico [1]. This allows one to partially substitute the expensive experimental screening with computational simulation, and thereby speed up and reduce the cost of developing new materials.

Nevertheless, NP-complexity of the problems in this field of research drastically complicates method development. That is why the prediction of new materials by random search is impossible. An effective global optimization algorithm can help to overcome this problem. Evolutionary algorithms, particle swarm optimization, random sampling, simulated annealing, and metadynamics are the most popular methods for this purpose [1].

In the current study, an adaptation of the USPEX software [2] to the BOINC [3, 4] distributed computing system interface is considered. USPEX is a well-known and widely used application, originally developed for the crystal structure prediction and is now capable of predicting structures of surfaces [6], two-dimensional crystals [7], nanoclusters [8] and polymers [9]. One of the reasons of the efficiency of USPEX is the combination of global and local optimization [10], which makes it possible to perform reliable global optimization even for quite complex systems.

However, for rigorous local optimization, a reliable method of structure relaxation and calculation of their energy (or any other property as a fitness parameter) is required. For these purposes, we have already interfaced the GULP code [11] with the BOINC system. Unfortunately, such calculations are of low accuracy and are only suitable as a crude approximation, because they are based not on quantum-mechanical methods, but on empirical force fields.

Because of this, most calculations using the USPEX algorithm are carried out using the Vienna Ab initio Simulation Package (VASP) [12], using density functional theory (DFT) [13, 14]. This software package allows performing ab-initio modeling of materials by approximate solution of the Schrödinger equation for periodic systems.

***Corresponding Author: Nikolay P. Khrapov:** Institute for Information Transmission Problems, Moscow, Russia; Email: nkhrapov@gmail.com
Valery V. Rozen: Moscow Institute of Physics and Technology, Moscow Reg., Russia
Artem I. Samtsevich: Skolkovo Institute of Science and Technology, Moscow, Russia
Mikhail A. Posypkin: Federal Research Center "Computer Science and Control", Moscow, Russia
Vladimir A. Sukhomlin: Moscow State University, Moscow, Russia
Artem R. Oganov: Skolkovo Institute of Science and Technology, Moscow, Russia; Stony Brook University, Stony Brook, United States of America; Moscow Institute of Physics and Technology, Moscow Reg., Russia

In VASP, the electronic wave functions are expanded in plane waves. Interactions between electrons and ions are described using ultrasoft pseudopotentials [15] or projector-augmented wave method [16, 17]. To determine the electronic ground state, VASP uses effective iterative matrix diagonalization methods such as residual minimization with direct inversion in the iterative subspace (RMM-DIIS) or a Davidson block iteration scheme. They are interconnected with highly efficient density mixing schemes of Broyden and Pulay to accelerate the self-consistency cycle.

The effective work of USPEX involves searching over a large number of crystal structures. With the help of the VASP software package adapted for work with the BOINC system, unique opportunities for computational materials design were opened.

The use of VASP requires a special license. The data files used by the VASP are also subject to intellectual property. Thus, it is necessary to protect the software and the data from unauthorized access.

When a BOINC-user connects his/her computer to the BOINC-project, the computer downloads the executable job file and the data files from the server. This approach assumes storing the application executable file and the data files on the file system of the volunteer computer. Thus, a user has instant access to this information and can obtain a pirate copy of the software. This is a great risk, which we have resolved in this work.

2 Related Work

There is an approach to virtualization provided by BOINC called VirtualBox-wrapper [18]. This approach assumes that the application is located inside the virtual machine image that runs on a volunteer computer. At the beginning of computations, the operating system is booted. With this approach, the operating system is not password protected. Thus one can extract the VASP executable from the virtual machine, thereby compromising the software privacy. A possible approach to protect the files is to store them in an encrypted folder. However this approach is impossible with the standard VirtualBox-wrapper which loads the OS each time the BOINC client is initialized. It implies that, each time the folder password should be supplied to the OS and therefore, it should be stored inside the image. Thus, the password and hence the executable file can be easily obtained.

A noteworthy approach is based on the use of partition encryption. This approach uses dm-crypt utility for Linux OS. The article [19] describes the application of the approach to hypervisors Xen and VMware. But BOINC is compatible only with VirtualBox hypervisor. Thus the approach is not compatible with BOINC.

Another important solution in the field of data concealment is steganography [20]. Steganography allows hiding data inside files of a different format. This approach involves deception of the user. Thus, it is unacceptable within volunteer computing.

The security mechanisms implemented in the hypervisor is considered in [21, 22]. This approach involves the modification of the VirtualBox engine. Thus, volunteers would have to install a modified VirtualBox software which seems to be hardly feasible.

3 Description of the proposed approach

In order to protect the information reliably, we developed a special wrapper. Our approach approved by VASP developers combines the capabilities of the BOINC-wrapper and of the VirtualBox-wrapper.

The main features of the proposed approach are as follows:

- The VM starts on the volunteer's computer from the saved state. Thus, application contains a paused virtual machine. Login to the system is not performed.
- The virtual hard disk image contains an encrypted partition. We use the AES symmetric encryption scheme. VASP executable is located inside this partition. A similar approach is described in [19]. The password or hash of the encrypted partition is stored into the RAM of the Virtual Machine. To enable the virtual machine, user does not need a password.
- The VASP data files are located inside the encrypted zip-archive. The zip-archive is located on the portable Virtual Disk Image (VDI) that is stored inside BOINC workunits. The password of the zip-archive is stored into the encrypted partition.

The application consists of several files:

- the vdi-disk with the operating system and the encrypted partition;
- the snapshot of the virtual machine;
- the vm-producer control script;
- BOINC-wrapper and the job.xml file for it.

USPEX algorithm generates input data and processes the output. For transportation to the client, the VASP input

files are placed inside the encrypted archive. The archive is stored in data.vdi disk image that is sent to the volunteer computer where the workunit's processing begins.

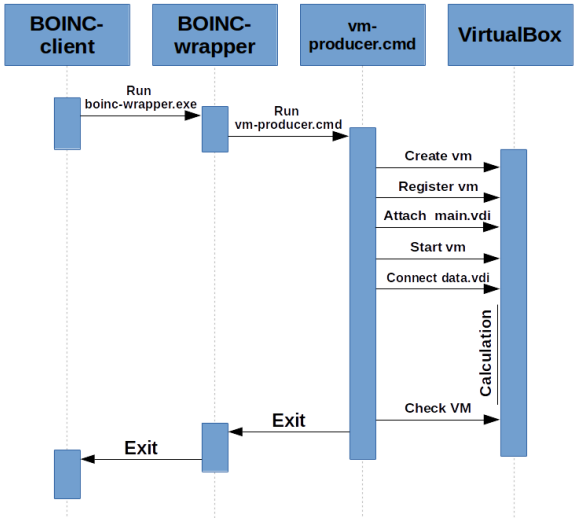

Figure 1: Workunit's processing

Figure 1 shows the sequence of the following actions:

1. The BOINC-wrapper application launches the control script vm-producer.cmd.
2. The control script performs the following actions:
   a) creates and registers new VirtualBox VM;
   b) attaches main.vdi to the VM;
   c) adopts state of the VM from the snapshot.sav;
   d) launches the VM from the saved state;
   e) attaches data.vdi to the VM;
   f) waits while the VM is running;
   g) unregisters the VM after it is turned off;
   h) terminates and transfers control to the BOINC-wrapper.
3. The BOINC-wrapper application exits thereby terminating the BOINC-task.
4. The data.vdi image with VASP output data is transferred to the server.

Running VASP is managed by the vasp-producer.sh script which takes the following actions:

1. waits until data.vdi is registered as an external SATA hard drive ("data-drive" in the sequel);
2. mounts the data-drive;
3. copies zip-file with VASP input files from data-drive to the encrypted disk;
4. decrypts zip-file;
5. launches the VASP application;

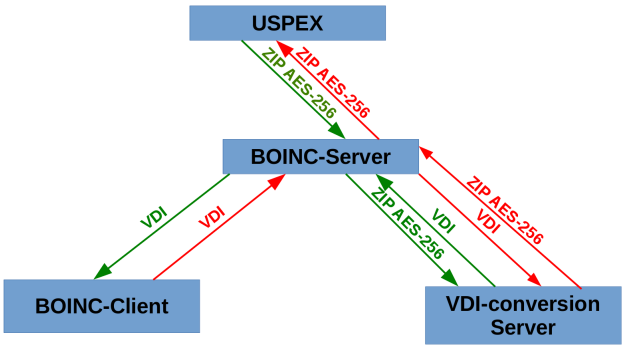

Figure 2: Computational infrastructure and the way of the USPEX task

6. encrypts results and copies them to the data-drive;
7. vasp-producer.sh shutdowns the operating system after VASP is done.

USPEX server splits the problem into many jobs. Each job undergoes the complex sequence of transformations shown in Figure 2 and described below.

1. USPEX-server aggregates several tasks into one archive. Then USPEX-server encrypts the archive and submits it to the BOINC-server.
2. BOINC server gets the archive from the USPEX-server, then BOINC-server submits it to the server of VDI-conversion. VDI-server puts the encrypted archive to the VDI-file. Then the job is returned to the BOINC-server.
3. Then the BOINC-server submits the task to the computational node.
4. The task is performed on the compute node. Workunit's processing on the computing node is described above. After completing the calculations the result is submitted to the BOINC-server. Result of the computation is located into the encrypted zip-file. This file is located into vdi-image.
5. The BOINC-server submits the job to the VDI-server to extract zip from VDI-file.
6. After VDI-conversion the BOINC server returns the job to the USPEX-server.
7. The USPEX-server decrypts the result and uses it.

4 Conclusions

We proposed a new approach to use proprietary software in a volunteer computing framework. The approach is based on the VirtualBox virtualization technology. The protection is provided by using encrypted folders and

password protected archives. The implementation of the approach makes it possible to provide the necessary level of security applicable to this particular case. Security mechanisms can be modified to fit a wide range of problems. It should be noted that the proposed approach involves some overhead due to the virtualization. However, experiments showed that this overhead is not significant. The implementation of the approach was successfully applied in USPEX@home project [23] aimed at computational materials discovery.

Acknowledgement: A.R.O. thanks Russian Science Foundation (grant 16-13-10459) for financial support. Mikhail Posypkin and Nikolay Khrapov were funded by the NSH-8860.2016.1 project of the governmental program for supporting leading scientific schools and RFBR project 16-07-00873 A.

References

[1] Oganov A.R. (Editor). Modern Methods of Crystal Structure Prediction. Berlin: Wiley-VCH. ISBN: 978-3-527-40939-6. (2010).

[2] Oganov A.R., Glass C.W., Crystal structure prediction using ab initio evolutionary techniques: principles and applications. J. Chem. Phys, 2006, 124, art. 244704.

[3] Anderson D.P., BOINC: A System for Public-Resource Computing and Storage, Proceedings of the 5th IEEE/ACM International Workshop on Grid Computing, IEEE Computer Society, Washington, DC, USA, 2004, 4–11, doi:10.1109/GRID.2004.14.

[4] Ivashko E.E., Nikitina N.N., Web service of access to computing resources of BOINC based desktopgrid. International Conference on Parallel Computing Technologies, Springer Berlin Heidelberg, 2011, 437-443.

[5] Zhang W.W., Oganov A.R., Goncharov A.F., Zhu Q., Boulfelfel S.E., Lyakhov A.O., Stavrou E., Somayazulu M., Prakapenka V.B., Konopkova Z., Unexpected stoichiometries of stable sodium chlorides, Science, 2013, 342, 1502-1505.

[6] Zhu Q., Li L., Oganov A.R., Allen P.B., Evolutionary method for prediction of surface reconstructions with variable stoichiometry. Phys. Rev., 2013, B87, 195317.

[7] Zhou X.F., Dong X., Oganov A.R., Zhu Q., Tian Y., Wang H.T., Semimetallic two-dimensional boron allotrope with massless Dirac fermions, Phys. Rev. Lett., 2014, 112, 085502.

[8] Matsko N. L., Tikhonov E. V., Baturin V. S., Lepeshkin S. V., Oganov A. R., The impact of electron correlations on the energetics and stability of silicon nanoclusters, J. Chem. Phys., 2016, 145, 074313.

[9] Sharma V., Wang C., Lorenzini R.G., Ma R., Zhu Q., Sinkovits D. W., Pilania G., Oganov A.R., Kumar S., Sotzing G.A., Boggs S.A., Ramprasad R., Rational design of all organic polymer dielectrics, Nat. Commun., 2014, 5, art. 4845.

[10] Oganov A.R., Lyakhov A.O., Valle M., How evolutionary crystal structure prediction works - and why. Acc. Chem. Res., 2011, 44, 227-237.

[11] Gale J.D., Rohl, A.L., The General Utility Lattice Program (GULP). Mol. Simul., 2003, 29, 291–341.

[12] Kresse G., Furthmüller J., Phys. Rev., 1996, B 54 11169.

[13] Hohenberg P., Kohn, W., Inhomogeneous Electron Gas, Phys. Rev., 1964, 136 B864.

[14] Kohn W., Sham L.J., Self-Consistent Equations Including Exchange and Correlation Effects, Phys. Rev. 1965, 140, A1133.

[15] Vanderbilt D., Soft self-consistent pseudopotentials in a generalized eigenvalue formalism. Phys. Rev., 1990, B 41, 7892(R).

[16] Blöchl P.E., Jepsen O., Andersen O. K., Improved tetrahedron method for Brillouin-zone integrations, Phys. Rev., 1994, B 49, 16223.

[17] Kresse G., Joubert D., From ultrasoft pseudopotentials to the projector augmented-wave method. Phys. Rev., 1999, B 59, 1758.

[18] VirtualBox Applications, https://boinc.berkeley.edu/trac/wiki/VboxApps, last accessed 2017/01/30.].

[19] Chunxiao Li, Anand Raghunathan, Niraj K. Jha. Secure Virtual Machine Execution under an Untrusted Management OS. IEEE 3rd International Conference on Cloud Computing. 2010. pp. 172-179.

[20] Steganography project, https://www.openstego.com/.

[21] D. Rangegowda, R.M. Fries, November 2009, Corralling virtual machines with encryption keys, U.S. patent 20090282266 A1.

[22] Xinbao Liu, July 2013, Host virtual machine assisting booting of a fully-encrypted user virtual machine on a cloud environment, U.S. patent 20130173900 A1.

[23] USPEX@HOME homepage, http://uspex-at-home.ru/, last accessed 2017/10/