A GPU enhanced approach to identify atomic vacancies in solids materials.

Joaquín Peralta[1], Claudia Loyola[1], and Sergio Davis[2].

1 Grupo de Simulación Cuántica en Materiales y Biosistemas, Facultad de Ciencias Exactas, Departamento de Física, Universidad Andrés Bello. http://www.matbio.cl

2 Grupo de Nanomateriales, Departamento de Física, Facultad de Ciencias, Universidad de Chile. http://www.gnm.cl

Universidad Andrés Bello

Abstract

Identification of vacancies in atomic structures plays a crucial role in the characterization of a material, from structural to dynamical properties. In this work we introduce a computationally improved vacancy recognition technique, based in a previous developed algorithm. The procedure is based in the use of Graphics Processing Unit (GPU) instead of Central Processing Unit (CPU), taking advantage of random number generation as well the use of a large amount of simultaneous threads as available in GPU architecture, improving the spatial mapping in the sample and the speed during the identification process of atomic vacancies. The results show that with this technique, efficiency is improved. Along with the above a reduction of required parameters in comparison with the original algorithm is presented. We show that only the lattice constant and a tunable overlap are enough as input parameters in the process, and are also highly related. A study of those parameters is presented, suggesting how the parameter choice must be addressed. Benchmarks were made using one standard CPU and GPU between the original code and the present work, revealing an improvement in the execution time.

Introduction

It has been shown that the identification of vacancies is fundamental to understand different materials properties, such as: i) electronic and mechanical behavior due to the presence of vacancies in oxide inter-metallic alloy interfaces; ii) the relevance of their migration near to melting temperature, which provide a relevant information on melting process; and iii) collapse of crystals, here simulations suggest a strong connection with ring-like atomic movement, due to vacancies, among others. A previous work (1) gives us a complete and well--guided process to identify a vacancy in a crystalline or amorphous structure, by the use of virtual spheres (VS).

This work proposes two main branches: first to redesign the algorithm by the use of GPU architecture which will improve the speed of the process; and reduce the number of input parameters, where the original method [1] call for at least three different parameters to be tuned. In this work we will use the NVIDA CUDA [2] version 6.0 to compile the GPU code. A good performance scaling is expected when it is possible to split the calculations in different sub-tasks, and assign these to each GPU thread. In this particular case, the algorithm is parallelizable mainly because we can simultaneously analyze different regions of space and thus perform the vacancy search more efficiently. Our technique will allow us to improve the analysis time. On the other hand we eliminate one tunable parameter and relate the others in a general way, based completely on the unit cell of the material under study.

Algorithm and Parameters

A recently developed method [1], provide us with a procedure for recognizing atomic vacancies based in the incorporation of virtual spheres (VS) of radius R_0, which are overlapped with another spheres (atoms) as is shown in Figure 1. In this work we keep the use of a VS as the main component in the atomic vacancy search. However the spatial mapping of the neighborhood of this VS is analyzed using GPU--based techniques. The use of the GPU allows for a more efficient analysis for a considerable large number of VS (and possible vacancies) in the space. The technique will allow us to avoid the use of additional refinements, such as simulated annealing minimization. The new technique is briefly summarized as follows:

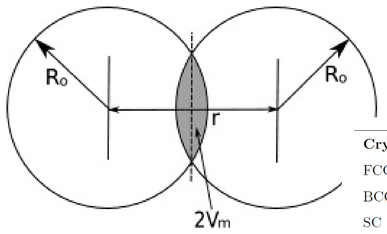

Figure 1 : The overlap between two spheres of radius R_0, with a distance *r* between the centers. Taken from (1).

Crystal	R_1	N_1	R_2	N_2	R_0	R_2/R_1
FCC	$a/\sqrt{2}$	12	a	6	$a(1/\sqrt{2}+1)/4$	$\sqrt{2}$
BCC	$\sqrt{3}a/2$	8	a	6	$a(\sqrt{3}+2)/8$	$2/\sqrt{3}$
SC	a	6	$a\sqrt{2}$	12	$a(\sqrt{2}+1)/4$	$1/\sqrt{2}$

- ***Sort original structure atoms by their overlap.***
- ***Build a cubic structure around an atom and generate uniformly distributed random points inside.***
- ***Using a large number of GPU threads, evaluate the overlap for each random point.***
- ***Search and found the minimum overlap value of the set of random points.***
- ***Identify if the point is a vacancy using the condition of overlap f_{ov}***

The choice of R_0, is suggested to be considered at some value between $R_1/2$ and $R_2/2$, that correspond to half the distances to first and second neighbor respectively. A value of R_0 close to $R_1/2$ will give us a larger number of vacancies, because it becomes most probable to find a lower overlap in some spatial points. On the other hand, if R_0 is close to $R_2/2$ the number of vacancies will be reduced because the total overlap will include partial contribution provided by second neighbors. An initial reasonable value for R0, used here, is the average value between the mentioned limits, $R_0 = (R_1+R_2)/4$.

In a perfect crystalline structure the expected overlap value of a missing atom is exactly related to the overlap of the number of closest neighbors. Despite the above the structure is not always related to a perfect crystal, because atoms are vibrating around an equilibrium position at finite temperature. For those cases, regarding displaced atoms from equilibrium positions, the value of r= R_1 must be modified. We will use the same R_1 value but with an additional percentage of displacement *x*. Using this simple approach to choose this parameter, we are able to define: $f_{ovp} = N\, f\left(\frac{4(1-x/100)}{1+R_2/R_1}\right)$.

Study case and results

As a test model of the algorithm we use a tungsten BCC structure of 2000 atoms at different temperatures, the samples were prepared from crystalline structures with a lattice constant of *a*=3.1652Å. The temperatures chosen are 300K, 1000K, 2000K, 3000K, 3500K and 6000K. Each sample is simulated during 50000 Δt with Δt =1 fs. The first 30000Δt a temperature control was used, by rescaling velocities. The last 20000Δt were simulated without any disturbance. The simulations were made using a standard molecular dynamics code(3), the inter--atomic potential used for tungsten was the proposed by Finnis--Sinclair.

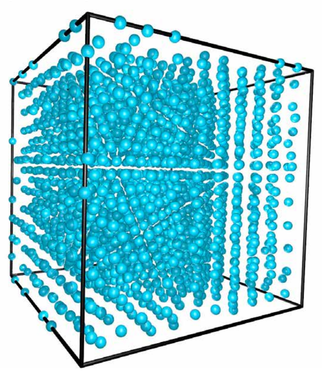

Figure 2 : A tungsten simulation cell at 1000 K, with a 2% of vacancies. During the preparation Periodic Boundary Conditions (PBC) were used.

FONDECYT Fondo Nacional de Desarrollo Científico y Tecnológico
Proyecto FONDECYT Iniciación 11130501

For each temperature we use an initial structure with 2000 atoms as observed in Figure 2, in a cubic cell of 31.652Å. For these structures, vacancies were randomly generated, ranged from 2% to 12% (each 2%) one hundred times. The final results are the average of those hundred structures analyzed with the algorithm.

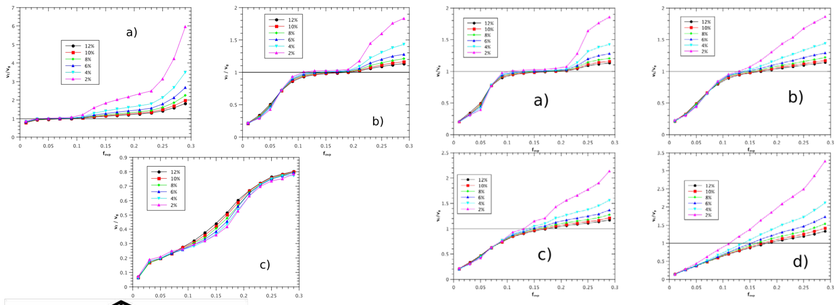

Figure 3 above: Results for different values of R0 and overlap fov.

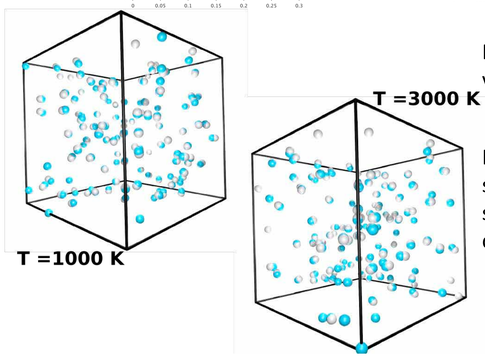

Figure 4 right: Generated (white spheres) versus found (light blue spheres) vacancies in samples at different temperatures.

Table 1: Speed comparative between CPU and GPU using the search vacancy algorithm

# of Atoms	2% GPU/CPU [s]	6% GPU/CPU [s]	12% GPU/CPU [s]
2000	6.50/75.23	7.60/95.42	8.9/160.1
4000	14.58/223.26	12.52/440.76	15.2/461.7
8000	37.93/1046.34	47.12/1555.72	58.5/2679.0
16000	118.00/4803.81	137.90/10769.61	177.3/13193.96
32000	422.01/30917.45	549.08/53870.07	746.1/92219.03

Conclusions

The use of GPU architecture in the vacancy recognition technique, to find vacancies in crystalline and non--crystalline structures, has been incorporated. The results show an improvement in both calculation speed and reduction of adjustable parameters. The growing of the GPU techniques and the use of consolidate arithmetic operation on GPU, allow us to obtain confident data in the process. An improvement was achieved up to 100X in the speed, which mean that problems that involve a very large number of atoms are easily treatable. As it was showed from the original algorithm, the use of this technique in different crystalline or amorphous structures is based in just the modification of the used parameters, which are highly related to the unit cell of the studied structure. Further treatment can be addressed to multi--component systems, as well as for systems that involve voids, as the detection of voids regions in a structure plays a relevant role in
materials properties.

References

(1) S Davis, A Belonoshko, and B Johansson. Computer Physics Communication 182(5), 1105-1110.
(2) J Nickolls, I Buck, M Garland, K Skadron. Scalable parallel programming with CUDA, Queue 6 (2) (2008) 40.
(3) S Davis, C Loyola, F Gonzalez, J Peralta. Computer Physics Communications 181 (12), 2126-2139.