

DIALOGUE BETWEEN MOLECULAR DYNAMICS AND
CONTINUOUS MEDIA: DEFINITION OF A COHESIVE
MODEL BASED ON ATOMIC SCALE INFORMATION

- 2nd YEAR INTERNSHIP REPORT -

Academic Year 2021-2022



INTERNSHIP SUPERVISOR
Noel JAKSE
0102030405
noel.jakse@grenoble-inp.org

Thibault MROZ
2nd Year - SIM

TABLE OF CONTENTS

Table of Contents

Remerciements	2
Glossary	3
List of Figures	4
Liste of Tables	5
1 Introduction	6
1.1 Context	6
1.2 Internship's Problematic	6
1.3 Laboratory Presentation	7
1.3.1 History	7
1.3.2 Research Groups	7
1.4 Report Outline	7
2 Hardware and Methods	8
2.1 Hardware and Software	8
2.1.1 LAMMPS	8
2.1.2 Ovito	9
2.1.3 Perseus GRICAD	9
2.2 Methods	10
2.2.1 The Theory	10
a) Fracture Mechanics	11
b) Molecular Dynamics	13
2.2.2 The Model used	14
Conclusion et Perspectives	16
Bibliographie	17
Annexes	18
A) run.oar script	19
B) input.file script example	19

Remerciements

Glossary

Adiabatic type of thermodynamic process that happens without transferring heat or mass between systems.. 12

Cohesive Zone fracture formation is regarded as a gradual phenomenon and separation of the crack surfaces takes place across an extended crack tip, or cohesive zone, and is resisted by cohesive tractions. 10, 11

Molecular Dynamic a computer simulation method for analyzing the physical movements of atoms and molecules.. 11–13

NPT corresponds to an isothermal-isobaric ensemble.. 12

NVE corresponds to a microcanonical ensemble.. 12

NVT corresponds to a canonical ensemble.. 12, 13

SSH a cryptographic network protocol for operating network services securely over an unsecured network.. 9

LIST OF FIGURES

List of Figures

2.1	LAMMPS operation	8
2.2	Final Operation Scheme	9
2.3	SSH cluster access schema	10
2.4	Simplified fracture scheme	11
2.5	Interatomic force \vec{F} [3]	11
2.6	Problem splitting	12
2.7	Simplified schema of a molecular dynamic program	13
2.8	Model used for simulations	15

LIST OF TABLES

List of Tables

1 — Introduction

1.1 Context

Technological advances in Artificial Intelligence and Machine Learning (MLOps) have led to the development of many tools to facilitate the life of users and methods to advance research. A very recent phenomenon in the world of research, Machine Learning allows to save a lot of computing time to perform large scale simulations but also to make some predictions.

There are several branches of Machine Learning. The most classical one is the one where a program is given a lot of data and it learns from this data. The program can then provide a prediction based on the input parameters. However, by doing this, we lose the physical sense (if the data are physical simulations or experiments). Another branch is to sort the input data according to what is more likely to happen. The physical meaning is then preserved but the program will only be able to provide an estimation of what could happen.

1.2 Internship's Problematic

This internship is part of a multiscale analysis of fracture and more precisely by conducting a dialogue in molecular dynamics and description in continuous medium. More precisely, the aim is to identify a cohesive zone model, representing the fracture mechanism at the continuous scale through a "stress-vector" - "opening" relation. This model will be identifiable following calculations in molecular dynamics which produce the numerical experiments.

An important part of the work is to conduct Molecular Dynamics simulations on a Silicon (Si) crystal for which the fracture occurs by cleavage. However, since Si has anisotropic elastic properties, it is expected that the fracture properties are also anisotropic. Therefore, simulations for different orientations between the crack plane and the crystalline symmetry planes will also have to be carried out. A systematic approach can be conducted. Nevertheless, the methodology associating Machine Learning and Molecular Dynamics is to be exploited in order to gain in calculation time.

Once the cohesive model is identified, it is then possible to study and predict the interactions between cracking and microstructure (in a polycrystal for example), as well as between crack and cavity.

It is a 100% digital project with a strong interest in simulation methods and Machine Learning.

1.3 Laboratory Presentation

1.3.1 History

The Laboratory of Science and Engineering of Materials and Processes is the result of the merger of three units on January 1, 2007. It is a joint research unit: CNRS, Grenoble-INP, and IESA. It brings together an average of 220 people including 56 researchers and teacher-researchers, 37 engineers, technicians and administrative staff, 60 PhD students, post-doctoral fellows, guests and trainees.

1.3.2 Research Groups

The Laboratory relies on four research groups that perpetuate the basic sciences in physics and physical chemistry, thermodynamics and kinetics, solid and fluid mechanics:

- **EPM** : Elaboration by Magnetic Processes
- **GPM2** : Physical and Mechanical Engineering of Materials
- **PM** : Metal Physics
- **TOP** : Thermodynamics, modeling, Process Optimization

This internship is placed between two divisions (PM and TOP) in a small team composed of :

- **Noel JAKSE** : Teacher-researcher in the TOP research group, Master and Supervisor of the internship
- **Rafael ESTEVEZ** : Researcher, Co-Supervisor of the internship
- **Thibault MROZ** : Intern Assistant Engineer

The TOP Research Group focuses on materials development, thermodynamic phenomena (stability and characterization) and atomistic, thermodynamic, kinetic and reactor modeling. This has applications in the fields of thin films, complex metal alloys and functional materials.

The PM Research Group focuses on the metallurgy of metals: atomic structure, mechanical and physical properties and oxidation. This has applications in the fields of materials for energy and microelectronics but also for structural materials.

The internship is in the field of atomistic modeling and atomic structure.

1.4 Report Outline

2 — Hardware and Methods

In this part, will be detailed all the hardware, software and methods used to carry out simulations.

2.1 Hardware and Software

As this internship is 100% digital, a good computer is required. A personal computer (MacBook Air M1) as well as a computer provided by the laboratory (under Ubuntu) will be the main equipment for this internship.

The main softwares are the following ones:

- **Visual Studio Code (VS Code)**: a source-code editor developed by Microsoft.
- **Large-scale Atomic/Molecular Massively Parallel Simulator (LAMMPS)**: a molecular dynamics programm (coded in C++) from Sandia National Laboratories.
- **Ovito**: a visualization and analysis software for output data generated in molecular dynamics.
- **Perseus GRICAD**: high performance computing and storage platforms.

2.1.1 LAMMPS

LAMMPS is an open-source molecular dynamic code with a focus on materials modeling. It provides potentials for solid-state materials (metals and semiconductors). It can be used to model atoms or, more generically, as a parallel particle simulator at the atomic, meso, or continuum scale.

LAMMPS does not have any graphic interface which makes the handholding not that easy. The input code is written in .txt files that are compiled through a `Makefile` called with a `bash` command : `lmp_serial -in input.file.txt`.

LAMMPS provides a `log.lammps` file as output. All the behaviour of the script (output values, warnings, errors ...) is written in this file. However, with specific commands, this software can provide other outfile such as a `dump.test` file, which will be useful to have a visualization of the material behaviour.

Here is a quick recap :

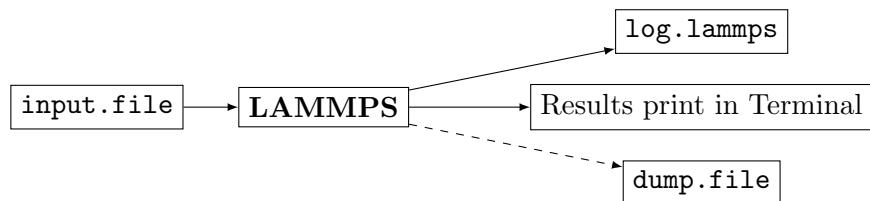


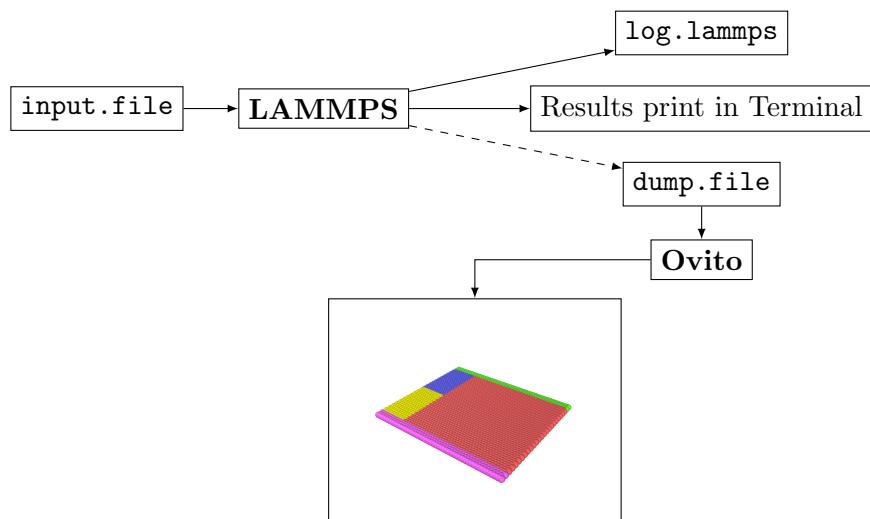
Figure 2.1: LAMMPS operation

2.1.2 Ovito

Ovito is a scientific visualization and data analysis tool for atomistic and other particle-based models. The community edition is free of charge under an open source license. Ovito has a Pro version which is a powerful extension with extended analysis toolset, visualization capabilities and automation with the Python integration. For this internship, the community edition is used.

Ovito will be used to visualize the behaviour of the atoms (mainly their position and velocity along the x,y and z axes). It will help to have a first sight of the simulation to see if there is no inconsistent behaviours before going deeper in the process.

The visualization is based on the `dump.file` that LAMMPS is producing. So here is the final scheme :

**Figure 2.2:** Final Operation Scheme

2.1.3 Perseus GRICAD

GRICAD offers intensive computing and data processing infrastructures to answer the needs of scientists. This tool provides an access to computing, grid, cloud, notebook and associated storage platforms. Moreover, an user support with assistance is opened. These infrastructures are open to all members of the scientific communities of the Grenoble site, as well as to their external collaborators. To have an access to this computing tool, a Perseus account is required. Once the Perseus account is created, you need to be member of the project to run your scripts. For this internship, the project is `pr-atosimul`.

GRICAD provides four computing clusters that are different (each cluster have their own hardware and configuration). Cluster access is normally done using a SSH Client (Secure Shell Protocol) [1]. However, SSH servers are vulnerable to scans and attacks so, for security reasons, it is not possible to let the clusters be directly accessed from the internet. GRICAD provides two SSH gateways that are more secure than the clusters. So the login method is to first, login to an SSH gateway and the login to the targeted cluster :

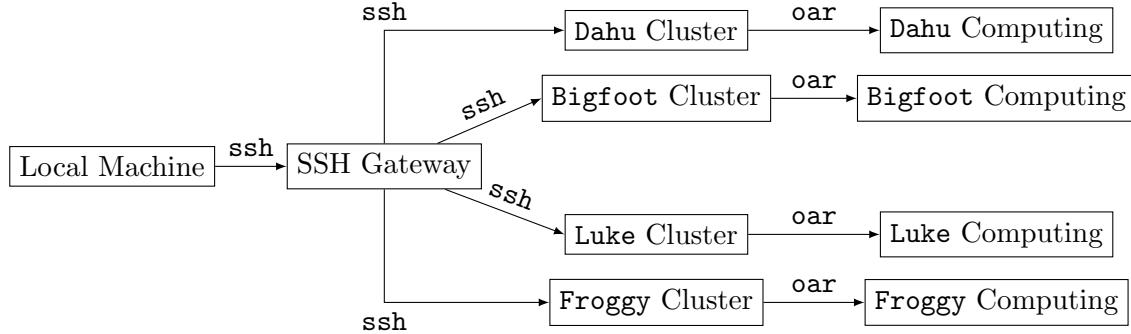


Figure 2.3: SSH cluster access schema

Those two SSH gateways (called **Rotule** and **Trinity**) are grouped under a single DNS : `access-gricad.univ-grenoble-alpes.fr`. This allows for load balancing on these two machines. Moreover, if one server came to fail, the other one is still available for computing.

The submission work for computing is made through a `run.oar` file. It is a `bash` script that provides the number of nodes and cores of the processor wanted by the user, the walltime (max time of computing), the name of some output files and then commands to run external scripts. A script is given in the [A](#) appendix.

Then, all the output files are stored on the cluster. To visualize them through Ovito for instance, a File Transfer Protocol Secure (FTPS) is used.

2.2 Methods

The aim is to make predictions about the behavior of cracks using a Machine Learning program. The algorithm will have to learn from the numerous simulations. It is therefore crucial that these input data are correct in order to limit the error on the predictions at the output of the program. So the first part of this internship is to work on the limit and initial conditions in order to have correct simulations.

2.2.1 The Theory

As mentionned before, this internship is a multiscale analysis. So, it focuses on Molecular Dynamic for the small scale and Fracture Mechanic for the large scale.

a) Fracture Mechanics

Here is a simplified scheme of a fracture :

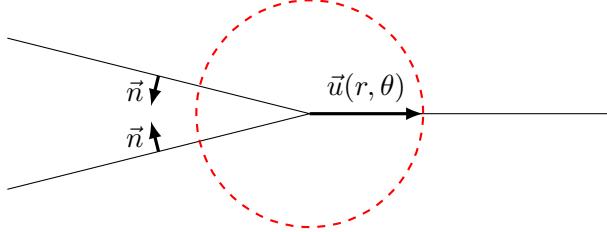


Figure 2.4: Simplified fracture scheme

A crack is defined by a discontinuity in the potential \vec{u} and that there is no stress on the free surfaces ($\vec{T} = \vec{0}$). According to Fracture Mechanic, the dashed red circle corresponds to the cohesive zone model. Outside this area, the surfaces are free of effort: $\vec{T} = \sigma \cdot \vec{n} = \vec{0}$. This is not the case in the cohesive zone where we have an interatomic potential $\bar{\vec{u}}$ defined as $\vec{u}(r, \theta) = f(r) \times g(\theta) \cdot \vec{u}_r$.

According to T.L. Anderson [2], locally (so when $r \rightarrow 0$), we have :

$$\vec{u} = \frac{K_1}{2\mu} \sqrt{r} f_{XY}(\theta) \quad \text{where } K_1 = \text{a loading parameter}$$

K_1 depends mainly on the type of crack (on the side, central, in tension, in traction...) and f_{XY} is a function that depends on X and Y which are the plan coordinates.

The interatomic potential induces an interatomic force $\vec{F} = \nabla \vec{u}$ which (*on the figure below, T is the stress vector defined as $\vec{T} = \sum_S \vec{f}_i$ where S corresponds to the surface*).

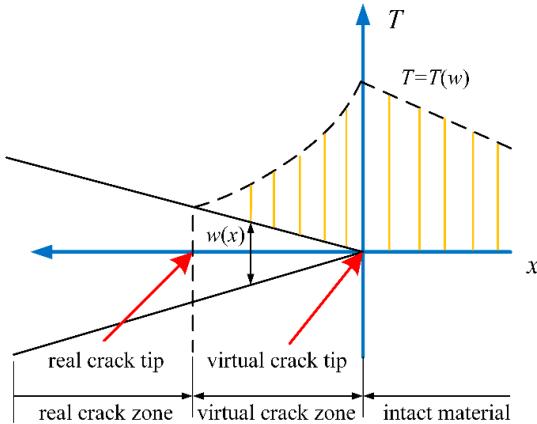


Figure 2.5: Interatomic force \vec{F} [3]

We can define a_0 which is the distance from the virtual crack tip where the interatomic force is zero : $\Delta = r - a_0$.

So, we have the following criteria :

$$\begin{cases} \Delta < 0 \Rightarrow f < 0 \Rightarrow \text{atoms repulsion} \\ \Delta > 0 \Rightarrow f > 0 \Rightarrow \text{atoms attraction} \end{cases}$$

This cohesive zone implies a difficult problem to resolve during a tensile test. The solution is to split the problem into two independent and simpler to solve problems. The first problem is a tensile test with a crack on the side without taking into account the cohesive zone model. The second problem is to take the cohesive zone model but without traction. To make the tests independent, there are two different loading parameters K_1 . In the case of the first test, it is a crack opening parameter $K_{1,opening}$. In the case of the second test, it is a crack closing parameter $K_{1,closing}$.

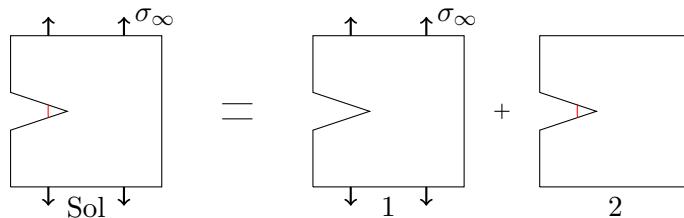


Figure 2.6: Problem splitting

Then, in order to calculate the crack propagation using fracture mechanics, a propagation criteria has been demonstrated by Alan Arnold Griffith [4] in his paper. He demonstrated for a tensile test charged with σ_∞ , with G the energy per unit of free area, A the free surface, W the crack work and a the crack length, that:

$$\frac{\partial W}{\partial A} = \frac{W[(2(a + da))] - W[(2a)]}{da} = 2\gamma$$

Moreover, $\frac{\partial W}{\partial A}$ corresponds to the consumed energy to create an additional free surface. So, as long as $G < \frac{\partial W}{\partial A}$, there is no crack propagation. However, when $G = 2\gamma$, there is propagation initiation.

That is how we calculate fracture propagation in materials according to fracture mechanics. But, we cannot really go deeper in the virtual crack zone and understand all the atomic interactions in the cohesive zone. To have a better understanding of potentials and crack propagation, we need to look closer and use molecular dynamic.

b) Molecular Dynamics

Molecular dynamics is a numerical method that resolver Newton's equation of motion $\vec{F} = \frac{d}{dt}(m\vec{v})$ for system of interacting. As the atoms are allowed to interact thanks to interatomic potentials for a fixed period of time, it gives the dynamic evolution of the system.

Here is a simplified schema of a molecular dynamic program (such as LAMMPS) :

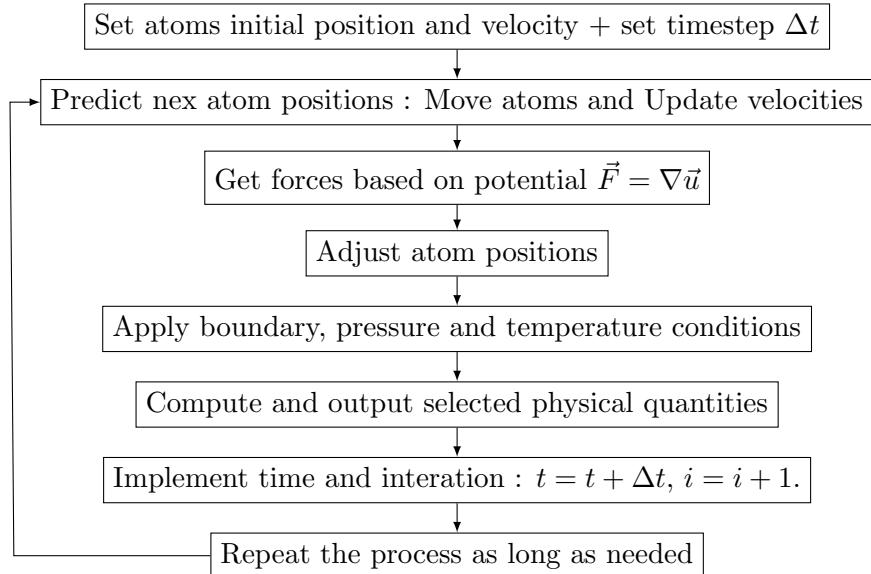


Figure 2.7: Simplified schema of a molecular dynamic program

The engineering of a molecular dynamic algorithm should account for the available computational power of the machine. That is why parameters such as timestep, simulation box size, number of atoms, potential... are wisely choose to have a correct computational time. Thoose simulatons requires sometimes high computational power to output some results. That is why a calculating computer is used during this internship. The type of system is also an important parameter to take into consideration. Only micro-canonical (NVE), canonical (NVT) and isothermal-isobaric (NPT) ensembles are used for simulations.

In the microcanonical ensemble, the system is totally isolated from changes in moles (N), volumes (V) and energy (E). It is linked with an adiabatic process without heat exchange.

In the canonical ensemble, the amount of atoms (N), volume (V) and temperature (T) are conserved. In a NVT system, the energy is exchanged with a thermostat. There are several thermostat algorithm and methods.

In the isothermal-isobaric ensemble, the amount of atoms (N), pressure (P) and temperature (T) are conserved. Compared to the NVT ensemble, a barostat is needed in addition to a thermostat. This ensemble is getting closer to laboratory conditions.

In addition to the choice of the type of system to run a simulation, molecular dynamics require a potential function which is a mathematical description of particles interaction. There are plenty of potentials that can be defined at many levels of physical accuracy. It exists pair potential functions in which the total potential energy can be calculated from the sum of all interatomic pairs energy. An example of such potential is the Lennard-Jones potential:

$$U(r) = 4\epsilon \left[\left(\frac{\sigma}{r} \right)^{12} - \left(\frac{\sigma}{r} \right)^6 \right]$$

A more precise one is the Stillinger-Weber one. It has been designed for Silicone. The mathematical is complicated and without interest for the internship. It is composed of two terms: a two-body term (which corresponds to the interaction energy between two neighboring atoms) and a three-body term (specifically added to energetically promote the tetrahedral environment of the atoms).

However, empirical potentials (often called force fields) are frequently used in material physics. Force fields potentials consist of a summation of bonded forces associated with chemical bonds, electrostatic charges and Van der Waals interactions. These potentials contain a huge amount of free parameters (atomic charge and radius, bond length and angles...) that makes calibration complicated. Its calculation is the bottleneck in the speed of molecular dynamic simulations. But, they are more precise than pair potential functions. An example of such potential is the ReaxFF potential.

2.2.2 The Model used

To simulate the propagation of a crack in a material, the model used is based on the theory of molecular dynamics. The idea is to carry out a small-scale tensile test on a sample with the beginning of a crack on the side. LAMMPS will allow to calculate the position, the speed and the forces exerted on each atom of the box.

Here is the model that will be used for simulations :

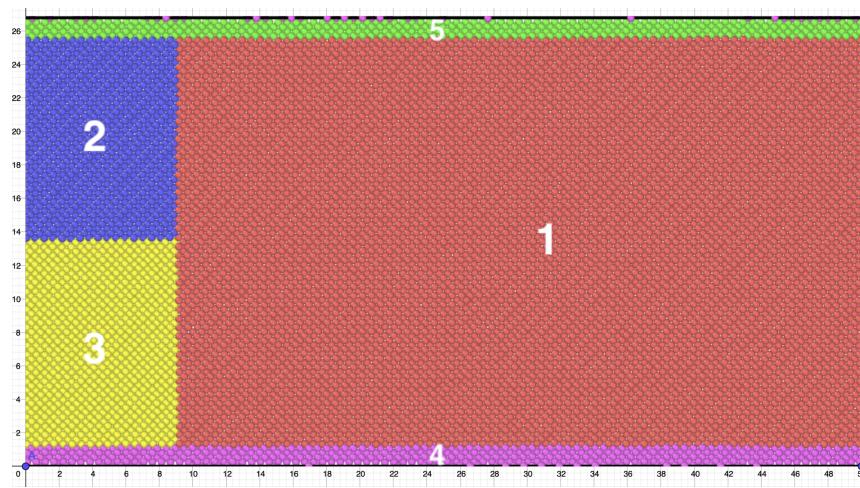


Figure 2.8: Model used for simulations

5 different regions are defined. There are 50 atoms along the x axis (horizontally) and 25 along the y axis (vertically). The dimensions of the box therefore vary according to the mesh parameter of the chosen material. During the tensile test, all regions are mobile except the 4th one. To simulate a tensile test in LAMMPS, we set the same velocity for all the mobile groups. To simulate the crack, regions 2 and 3 are not linked. An `input.file` is given in the B) appendix.

Conclusion et Perspectives

Bibliography

- [1] **Article**, Secure Shell
https://en.wikipedia.org/wiki/Secure_Shell
- [2] **Book**, T.L. Anderson, *Fracture Mechanics : Fundamentals and Applications*, Boca Raton, 2017
- [3] **Article**, Kaida Dai, Baodi Lu, Pengwan Chen and Jingjing Chen, *Modelling Microstructural Deformation and the Failure Process of Plastic Bonded Explosives Using the Cohesive Zone Model*, 2019
<https://www.mdpi.com/1996-1944/12/22/3661>
- [4] **Book**, A.A. Griffith, *The Phenomena of Rupture and Flow in Solids*, Royal Aircraft Establishment, 1920

Annexes

A) run.oar script

```

#!/bin/bash

folder=$(pwd | cut -d "/" -f 4)

#OAR -n lammps-reaxff-0,0
#OAR -l /nodes=1/core=32,walltime=10:00:00
#OAR --stdout lammps.%jobid%.out
#OAR --stderr pytest.%jobid%.err
#OAR --project pr-atosimul

# load environment
source /applis/site/guix-start.sh

# lancement du code

mpirun -np `cat $OAR_FILE_NODES|wc -l` lmp -in in_older.$folder-0,0.txt

```

B) input.file script example

```

# 2d LJ crack simulation

dimension      2
boundary       s s p

atom_style     atomic
neighbor       0.8 bin
neigh_modify   delay 10

# create geometry

lattice        hex 0.93
region         box block 0.0 50.0 0.0 25.0 -0.5 0.5
create_box     5 box
create_atoms   1 box

mass           1 1.0
mass           2 1.0
mass           3 1.0

```

ANNEXES

```

mass           4 1.0
mass           5 1.0

# LJ potentials

pair_style      lj/cut 2.5
pair_coeff      * * 1.0 1.0 2.5
pair_modify     shift yes

# define groups

region          1 block INF INF INF 1.0 INF INF
group           lower region 1
region          2 block INF INF 24.0 INF INF INF
group           upper region 2
group           boundary union lower upper
group           mobile subtract all boundary

region          leftupper block INF 10.0 12.5 INF INF INF
region          leftlower block INF 10.0 INF 12.5 INF INF
group           leftupper region leftupper
group           leftlower region leftlower

set             group leftupper type 2
set             group leftlower type 3
set             group lower type 4
set             group upper type 5

# initial velocities

compute         new mobile temp
velocity        mobile create 0.001 887723 temp new mom yes rot yes
velocity        upper set 0.0 0.003 0.0
velocity        mobile ramp vy 0.0 0.003 y 1.0 24.0 sum yes

# fixes

fix            1 all nvt temp 0.01 0.01 1.0
fix            2 boundary setforce NULL 0.0 0.0

# run - NE PAS TOUCHER AU Timestep

```

```
timestep          0.001
thermo            100
thermo_style     custom step temp pyy lx ly lz
thermo_modify    temp new

neigh_modify     exclude type 2 3

dump              1 all custom 100 dump.test id type x y z fx fy fz

run               10000
```