Industrial Training **Report**
Student: Eric Neiva 25th January 2016

1 Introduction

This document describes the main tasks carried out during the internship at the Structural Mechanics (SM) group at CIMNE Barcelona, under supervision of professor Michele Chiumenti, and at the Large Scale Scientific Computing group (LSSC) at CIMNE Castelldefels, under supervision of professor Santiago Badia.

The main purpose of the internship was to take the first steps to transform a serial code for the thermo-mechanical simulation of metal deposition processes into a parallel one. The outcomes of this work will be of use for the CIMNE participation in several related research projects, such as CAxMan, and are part of another ambitious project: the Master and PhD theses of the student.

2 Summary of tasks

A brief description of each task carried out is given in this section. Table 1 corresponds to tasks done at the SM group with COMET, the in-house software for thermo-mechanical problems, whereas Table 2 corresponds to tasks developed at the LSSC group with FEMPAR, the in-house software to solve multi-physics problems with parallel solvers.

TASK 1: Literature review	
Contents	**Results**
1) State-of-the-art in modelling metal deposition, welding, and casting processes at the SM group. 2) Assessment of applications, impact and prospects of Additive Manufacturing (AM) in the industry.	1) Understand the physics and the finite-element framework behind the numerical simulation of metal deposition processes. 2) Organize the information for the introductory chapters of the theses and for pitches of the topic in oral presentations.
TASK 2: Learn to use COMET	
Contents	**Results**
1) Design and run a series of examples to explore all the capabilities of the software using the three linked GiD problem types. 2) Reproduction of some examples of the literature review and test the influence of some parameters.	1) Learn how to create a numerical simulation of a metal deposition process and present its results (e.g. with videos). 2) Gain some knowledge about modelling metal deposition processes that is not included in the literature review.
TASK 3: Exploration of the COMET code	
Contents	**Results**
1) Learn how the code is structured and identify the parts of the code that concern the simulation of metal deposition processes.	1) The student will use these subroutines as a staring point to develop the parallel implementation.
TASK 4: Compile COMET for UNIX platforms	
Contents	**Results**
1) Write the `Makefiles` to compile COMET for UNIX platforms, automatically generating the list of files to be included (*nearly finished*).	1) The SM group will gain a tool to automatically update the COMET executable for UNIX platforms.

Table 1: Tasks carried out at the SM group with prof. M.Chiumenti.

TASK 4: Enter the FEMPAR community	
Contents	**Results**
1) Set up the personal computer to use FEMPAR. 2) Review of the principles of object-oriented programming, version control and FEMPAR own coding style.	1) Introduction to FEMPAR as a developer. 2) Run some simple examples with the code.

TASK 5: Solve multimaterial poission problems with FEMPAR	
Contents	**Results**
1) Develop a module to solve poisson problems in a domain with piecewise constant viscosity coefficients.	1) Achieved first step to tackle the numerical simulation of metal deposition processes with a parallel framework.

TASK 6: Obtain FEMPAR input data from COMET GiD problem-types.	
Contents	**Results**
1) Write the template files needed to generate the input data for FEMPAR from the COMET problem-types.	1) To facilitate the validation of the parallel implementation, the input data for COMET and FEMPAR will be generated from the same GiD problem-type.

Table 2: Tasks carried out at the LSSC group with prof. S.Badia.

3 Personal review:

My experience with the internship is quite satisfactory. At first, I was a little overwhelmed by the amount of previous tasks I had to carry out just to get used to both environments at the SM group in Barcelona and the LSSC group in Castelldefels.

COMET and FEMPAR are both written in Fortran but are thought in a completely different way. COMET is a standard procedural code developed by mainly one person, whereas FEMPAR is a highly abstract and collaborative large scale object-oriented code.

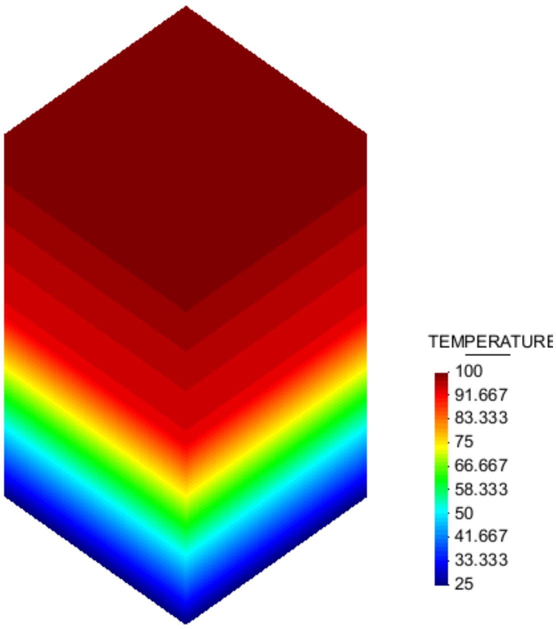
COMET relies on GiD as a pre- and post-processor. FEMPAR, on the other hand, uses pre- and post-processing tools thought for parallel environments, such as Paraview. It has its own GiD problem-type, but it only allows to generate serial numerical experiments.

The SM group uses Visual Studio to edit and compile COMET for Windows platforms. The FEMPAR community only works in UNIX platforms. The code is locally edited and analysed with eMacs or Understand and is shared and tested with a GitLab platform.

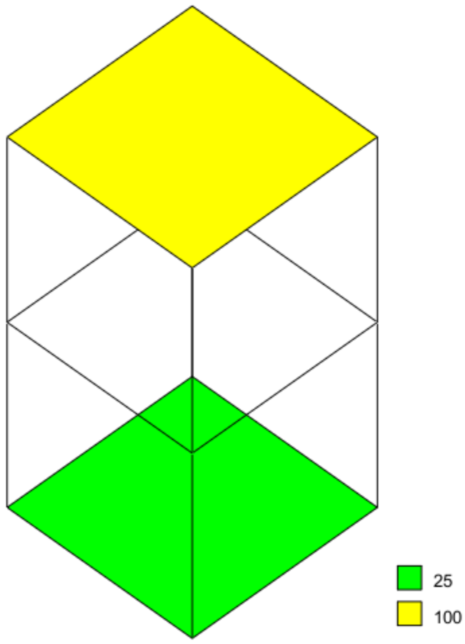
Fortunately, thanks to weekly meetings with M.Chiumenti and S.Badia, I was able to get through all these preliminary tasks in a relatively few time. Now I can start to tackle the real topics of my future research and I am really looking forward to it.

4 Selection of simulations

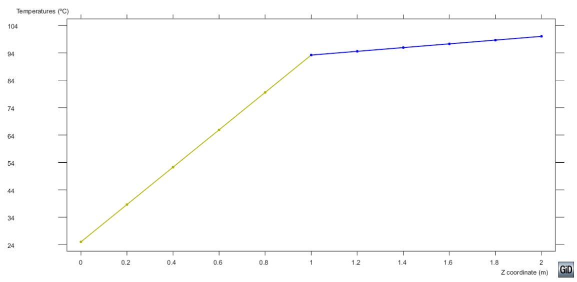
4.1 COMET → FEMPAR migration



(a) Temperature field contour



(b) Dirichlet boundary conditions


(c) COMET: Temperature along $z-$axis

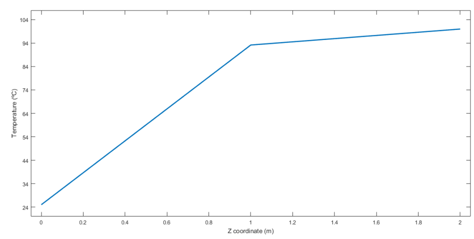
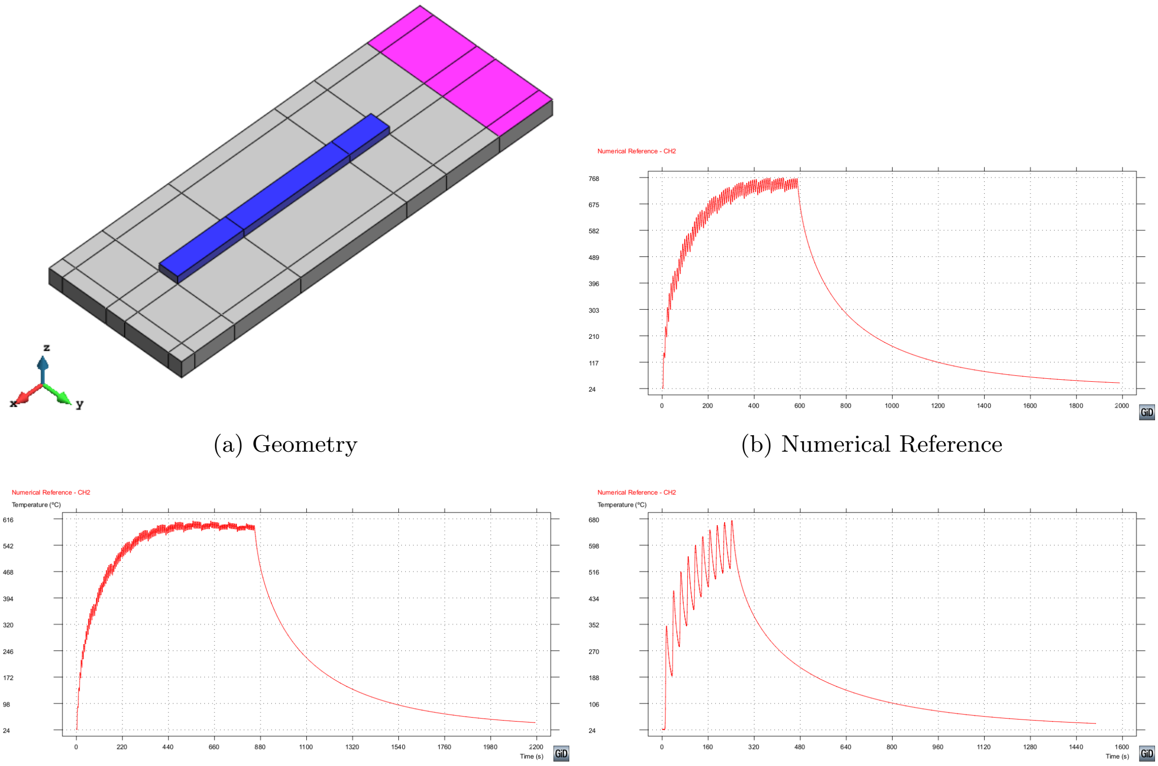

(d) FEMPAR: Temperature along $z-$axis

Figure 1: Simulation of a bimaterial poisson problem. $k = 1$ at the lower half of the prism, whereas $k = 10$ at the upper half of the prism. Homogeneous Neumann boundary conditions apply at the lateral walls. COMET and FEMPAR give exactly the same results.

4.2 Metal deposition processes

4.2.1 10 Layers - 7 Hatches (Example in Literature)



(a) Geometry

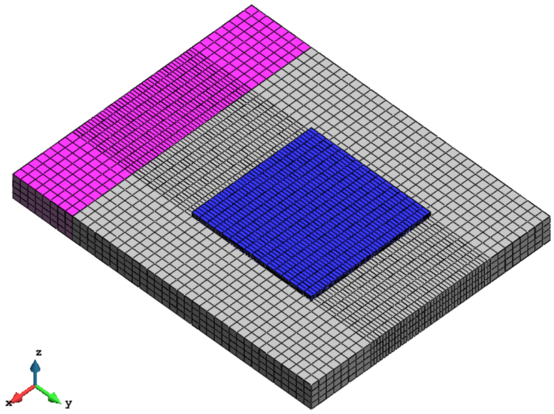
(b) Numerical Reference

(c) Check-patterned scanning sequence

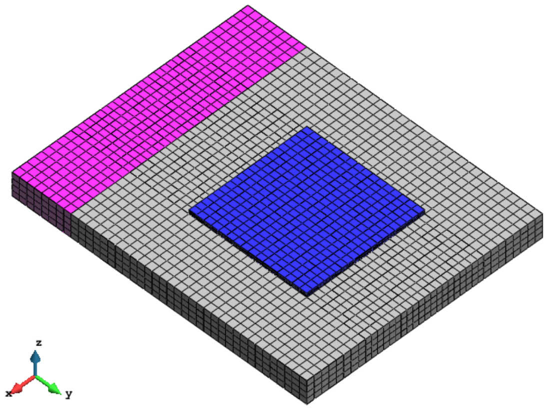
(d) Transverse scanning

Figure 2: Different scanning paths than the reference one were tested

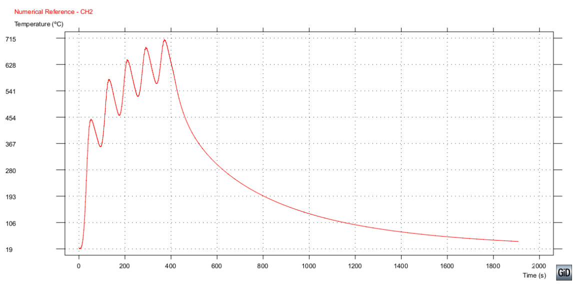
4.2.2 5 Layers - 10 Hatches



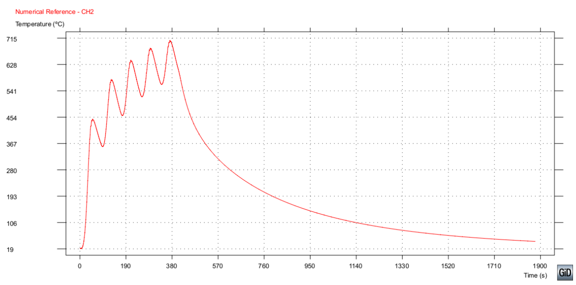
(a) Fine mesh



(b) Coarse mesh



(c) Temperature evolution for fine mesh



(d) Temperature evolution for coarse mesh

Figure 3: The influence of the discretization was also tested

4.3 More simulations

Refer to the two attached files.