Some mathematical models and numerical simulation of heat treatment of steel

VII Modelling Week 2013
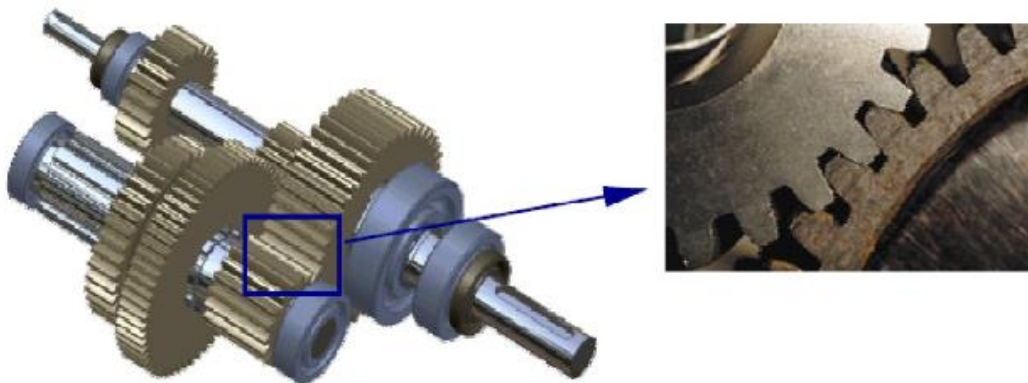
Group 5: Irene Díaz Bravo, Madeleine Kotzagiannidis, Florencia Rotondi, Alicia Tejada Torres

26/06/2013

1. Introduction

In the automotive industry there are many important moving pieces such as gears, wheels or racks and pinions which are in close contact in order to transmit a desired rotation or translation.

Some of these workpieces need to have a lifetime of at least 20 years. They are made of steel and that is why we need to give them a specific treatment in order to optimize their hardness and performance.

We need to know that before any hardening process steel is a ductile material. Work pieces during their lifetime are subject to some stresses, this explains why giving them a hardening treatment is so necessary.

Hardening treatment is applied to produce a hard boundary layer to prevent wear and abrasion and a soft inner part to reduce fatigue.
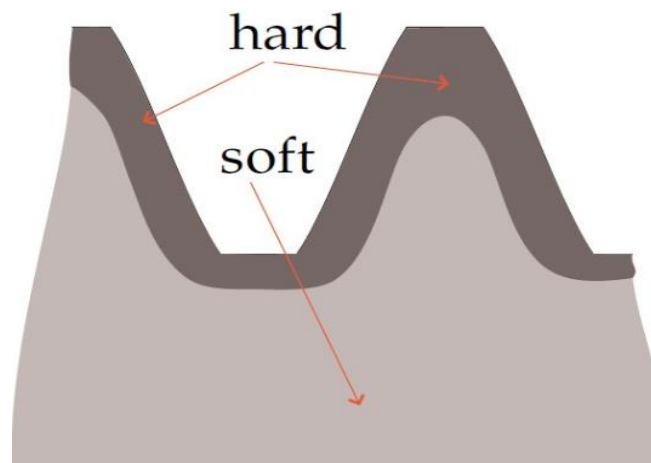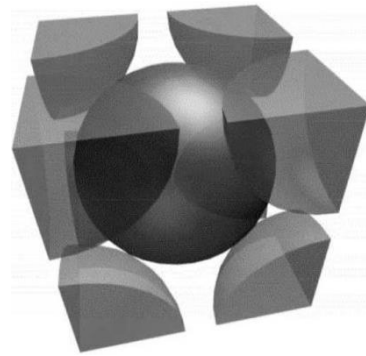
Steel comes from an iron based alloy which may appear in two different crystal lattices. The two types are fcc and bcc, we can see that the first one is denser and more compact.
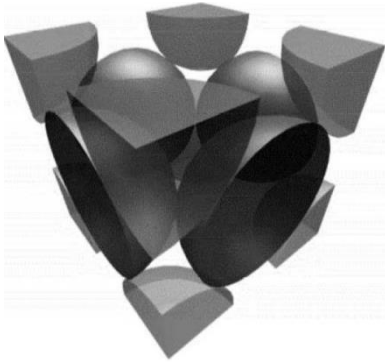
fcc (face centered cubic)   bcc (body centered cubic)

Moreover, steel has different solid phases. Some of them are

- Austenite
- Ferrite
- Perlite
- Martensite

They have different physical properties, for example perlite is the softest of them all and martensite de hardest.

It is only possible to produce austenite if the concentration of carbon atoms is up to 2.11% and at high range of temperature.

In the next diagram we can see that we can obtain austenite by heating for example ferrite and by cooling austenite we obtain ferrite among others.

The material we get after cooling depends on the cooling strategy. For example, if we cool austenite too fast we will obtain Martensite and if we cool it slowly we will obtain Perlite.

Now, we are going to present some boundary value problems in 2D and 3D together with the free software packages (FreeFEM++) we have used in our simulations.

2. Elliptic Linear Equations. Two dimensional example

As we know, the general form of Elliptic Linear Equation is:

$$-\nabla[A(x)\nabla u] + \nabla[b(x)u(x)] + c(x)u(x) = f(x)$$

Where A is a positive defined matrix

$$-\nabla[A(x)\nabla u] = \sum_{i,j=1}^{N} \frac{\partial}{\partial x_i}[a_{ij}(x)\frac{\partial}{\partial x_j}]$$

When $A(x) = I$, then it will be the known equation, Laplace eq.

An easy example about two dimensional problems with Elliptic Linear Equations and boundary conditions

At first, we have to define our domain, Ω. That will be a circle with an ellipse, Ω2, inside like in the next picture. The circle without Ω2 will be denoted Ω1.
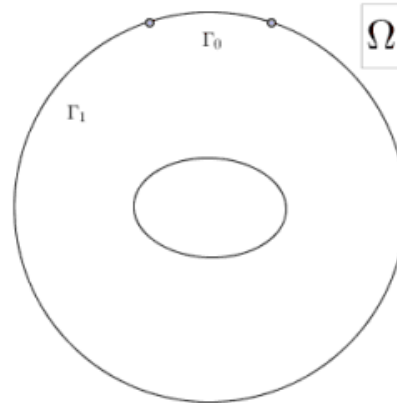
In this case, we assume that the material inside and outside the ellipse are different, and therefore, the diffusion coefficients will not be equal. So, we take k_1 as diffusion coefficient of circle and k_2 for the ellipse.

$$k(x) = \begin{cases} k_1 & in\ \Omega_1, \\ k_2 & in\ \Omega_2. \end{cases}$$

As we can see in the last picture, the boundary is defined by Γ_0 and Γ_1 where Γ_0 is the circle in $[\pi/4, 3\pi/4]$ and Γ_1 is the rest of the circle.
We will have a stationary problem with Dirichlet boundary condition on Γ_0 and Robin condition on Γ_1.

$$(P) \quad \begin{cases} -\nabla \cdot [k\nabla u] = f & in\ \Omega, \\ u = g & on\ \Gamma_0, \\ k\dfrac{\partial u}{\partial n} = \alpha(u_{ext} - u) & on\ \Gamma_1. \end{cases}$$

Where α is the transfer coefficient, f the source of energy, g the source that we apply in the boundary Γ_0, $(u_{ext}\text{-}u)$ is the different of temperature between the interior and exterior of Ω.

Now, the first step to solve this problem, we will generate the variational formulation of this problem. This change is essential in order to solve the problem using the software FreeFEM++.
 FreeFEM++ is the free software that we have used in all resolutions, getting easily the graphics that represent the evolution of heat on different surfaces in each of our problems.
Then, as we have said, at first, we have to compute the variational formulation of the problem P.
Let *v* a regular function such that $v : \overline{\Omega} \to \Box$ vanished on Γ_0. If we have

$$\int_{\Omega} -\nabla \cdot [k\nabla u] \cdot v = \int_{\Omega} f \cdot v$$

Integrating by parts in Ω, we have:

$$\int_{\Omega} k\nabla u \nabla v - \int_{\partial\Omega} k\frac{\partial u}{\partial n}v = \int_{\Omega} f \cdot v$$

$$\int_{\partial\Omega} k\frac{\partial u}{\partial n}v = \int_{\Gamma_0} k\frac{\partial u}{\partial n}v + \int_{\Gamma_1} k\frac{\partial u}{\partial n}v = \int_{\Gamma_1} \alpha(u_{ext} - u)v$$

Thus, u=g on Γ_0

$$\int_{\Omega} k\nabla u \cdot \nabla v + \int_{\Gamma_1} \alpha u \cdot v = \int_{\Omega} f \cdot v + \int_{\Gamma_1} \alpha u_{ext} \cdot v$$

For all regular function v such that *v*=0 on Γ_0.

Defining $V = \left\{ v \in L^2(\Omega) : \int_{\Omega} v'(x)v'(x)dx < \infty, v_{\Gamma_0} = 0 \right\}$

Now, we have to look for the expression of *u∈V* such that

$$\int_{\Omega} u'(x)v'(x)dx = \int_{\Omega} f(x)v(x)dx, \qquad \forall v \in V$$

We will denote

$$a(u,v) := \int_{\Omega} u'(x)v'(x)dx$$

$$L(v) := \int_{\Omega} f(x)v(x)dx$$

Then, the variational formulation of our problem is:

(VF)= { To find u∈V such that a(u,v)=<L,v> ,for all v∈V. }

Using Lax-Milgram theorem will explain existence and uniqueness of the solutions.
Theorem (Lax-Milgram):

Let V(.,.) an Hilbert space, a(.,.) a function from VxV to R a coercitive and bilineal form and L(.) another real and continuous function from V to R, then, the problem has solution and it is unique.   #

Our problem is stationary and for this reason, we do not have to discretize on time. Now, we already have the variational formulation of our problem and we just need introduce this formulation in FreeFEM++ software.

The main code that solves the problem is:

```
problem diffuDR(u,v) =
        int2d(Th)(kdiffusion*(dx(u)*dx(v)+dy(u)*dy(v)))
        + int1d(Th,Gamma1)(alpha*u*v)
        -int1d(Th,Gamma1)(alpha*uext*v)
      -int2d(Th)(Fsource*v) + on(Gamma0,u=0.);
```

Executing our FreeFEM++ program, we obtain next graphics:

Temperature Dirichlet-Robin
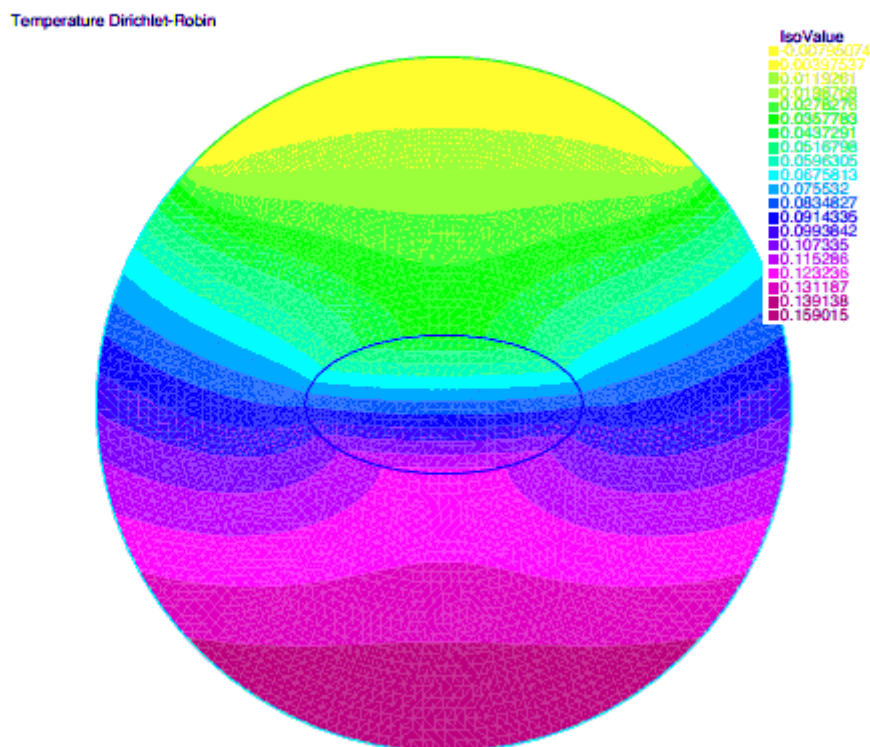
In this picture we can see the different behavior of heat depending of the domain Ω_0 or Ω_1. We can see that the heat is higher in Γ_0, where the heat source acts. The temperature decreases as we move away from the part of Γ_0. Furthermore, it is obvious that heat does no progress equally in Ω_0 and in Ω_1, and this is caused by having two different diffusion coefficients in Ω.

These are the result obtained of a stationary problem with Dirichlet and Robin conditions in the boundary. Now, we will see the way to solve another example, but no stationary, such that we will have to do a discretization on time.

Usually, we have to solve problems like (VF) when DimV=+ ∞. Galerkin methods give an approximation u_h to u.

For $d>1$, $d \in \square$, we consider a finite dimension space V_h with Dim $V_h = d$

$$V_h \subset V, \qquad V = \{\phi_1, \phi_2, ..., \phi_d\}$$

such that

$(VF)_h=$ To find $u_h \in V_h$ such that $a(u_h, v_h)=<L, v_h>$,for all $v_h \in V_h$

In this way, we have that $(VF)_h$ is equivalent to a linear system $A_h U_h = F_h$ for all $A_h \in U_{dxd}$, $F_h \in R^d$. It is to say:

$(VF)_h=$ To find $u_h \in V_h$ such that $a(u_h, \phi_i)=<L, \phi_i>$ with $1 \leq i \leq d$

We will explain the way to solve this type of problem using an example about an evolution case with heat equation.

$$\begin{cases} \rho c_0 u_t - \nabla \cdot [k\nabla u] = f & in\ \Omega \\ u = g & on\ \Gamma_0 \\ \dfrac{\partial u}{\partial n} = \alpha(u_{ext} - u) & on\ \Gamma_1 \\ u(x, 0) = u_0 & in\ \Omega \end{cases}$$

Where the domain Ω, the heat sources f and g are the same that in the previous example, and u_0 is the value of the function u in $t=0$.

At first, we need to apply a discretization on time, and in this case we are going to use Crank Nicolson Scheme.

Let M>1 integer, then $\Delta t=T/M$ and $t_m=n. \Delta t$, n=0,..,M. We denote $u^n(x) = u(x, n\Delta t)$.

Applying Crank Nicolson scheme we have:

$$\rho c_0 \frac{u^{n+1} - u^n}{\Delta t} - \nabla[k\nabla(\frac{u^{n+1} + u^n}{2})] = \frac{f^{n+1} + f^n}{2} \ \ in\ \Omega,$$

$$\frac{u^{n+1} + u^n}{2} = \frac{g^{n+1} + g^n}{2} \ \ on\ \Gamma_0,$$

$$\frac{\partial}{\partial n}\left[\frac{u^{n+1}+u^n}{2}\right] = \alpha\left(u_{ext} - \left(\frac{u^{n+1}+u^n}{2}\right)\right) \text{ on } \Gamma_1$$

$$u^0 = u_0$$

n=0,..M-1

Defining $w^{n+1} = \dfrac{u^{n+1}+u^n}{2}$ and replacing we have:

$$\rho c_0 \frac{w^{n+1}-u^n}{\Delta t/2} - \nabla[k\nabla w^{n+1}] = \frac{f^{n+1}+f^n}{2} \text{ in } \Omega,$$

$$w^{n+1} = \frac{g^{n+1}+g^n}{2} \text{ on } \Gamma_0,$$

$$\frac{\partial}{\partial n}[w^{n+1}] = \alpha(u_{ext}-w^{n+1}) \text{ on } \Gamma_1$$

$$\rho c_0(w^{n+1}-u^n) - \left(\frac{\Delta t}{2}\right)\nabla[k\nabla w^{n+1}] = \Delta t\left(\frac{f^{n+1}+f^n}{4}\right) \text{ in } \Omega,$$

$$w^{n+1} = \frac{g^{n+1}+g^n}{2} \text{ on } \Gamma_0,$$

$$k\frac{\partial}{\partial n}w^{n+1} = \alpha(u_{ext}-w^{n+1}) \text{ on } \Gamma_1$$

In this way, we obtain the next expressions:

$$\rho c_0 w^{n+1} - \left(\nabla\left[\frac{k\Delta t}{2}\nabla w^{n+1}\right]\right) = F^{n+1} \text{ in } \Omega,$$

$$w^{n+1} = G^{n+1} \text{ on } \Gamma_0$$

$$k\frac{\partial}{\partial n}[w^{n+1}] = \alpha(u_{ext}-w^{n+1}) \text{ on } \Gamma_1$$

Where

$$F^{n+1} = \frac{\Delta t(f^{n+1}+f^n)}{4} + \rho c_0 u^n$$

$$G^{n+1} = \frac{g^{n+1}+g^n}{2}$$

$$u^{n+1} = 2w^{n+1} - u^n$$

When the discretization is already applied, we have distinct problems for each t; a set of systems that we solve iteratively for looking for the final solution.

The way to solve these different problems depending the value of t is the same that before. We have to compute the variational formulation and introduce that in our FreeFem++ code. It will be like:

```
problem heat(u,v) = int2d(Th)(rho*cspecheat*u*v)
        + int2d(Th)(kdiffusion*deltat2*(dx(u)*dx(v)+dy(u)*dy(v)))
        + int1d(Th,Gamma1)(deltat2*alpha*u*v)
        -int1d(Th,Gamma1)(deltat2*alpha*uext*v)
      -int2d(Th)(Fsource*v) + on(Gamma0,u=GDirichlet);

func uinit = 273.; // Initial temperature

uaux = uinit;

 for(real t=0;t<=Tfinal;t+=deltat)
 {
 Fsource = deltat*(fsource(x,y,t+deltat)+fsource(x,y,t))/4.
                +rho*cspecheat*uaux;
 GDirichlet = (gDirichlet(x,y,t+deltat)+gDirichlet(x,y,t))/2.;

 heat;
 un1 = 2*u - uaux;
 uaux = un1;
 plot(un1,cmm="Temperature t="+t, fill=1, value=1, wait=0);
 }
```
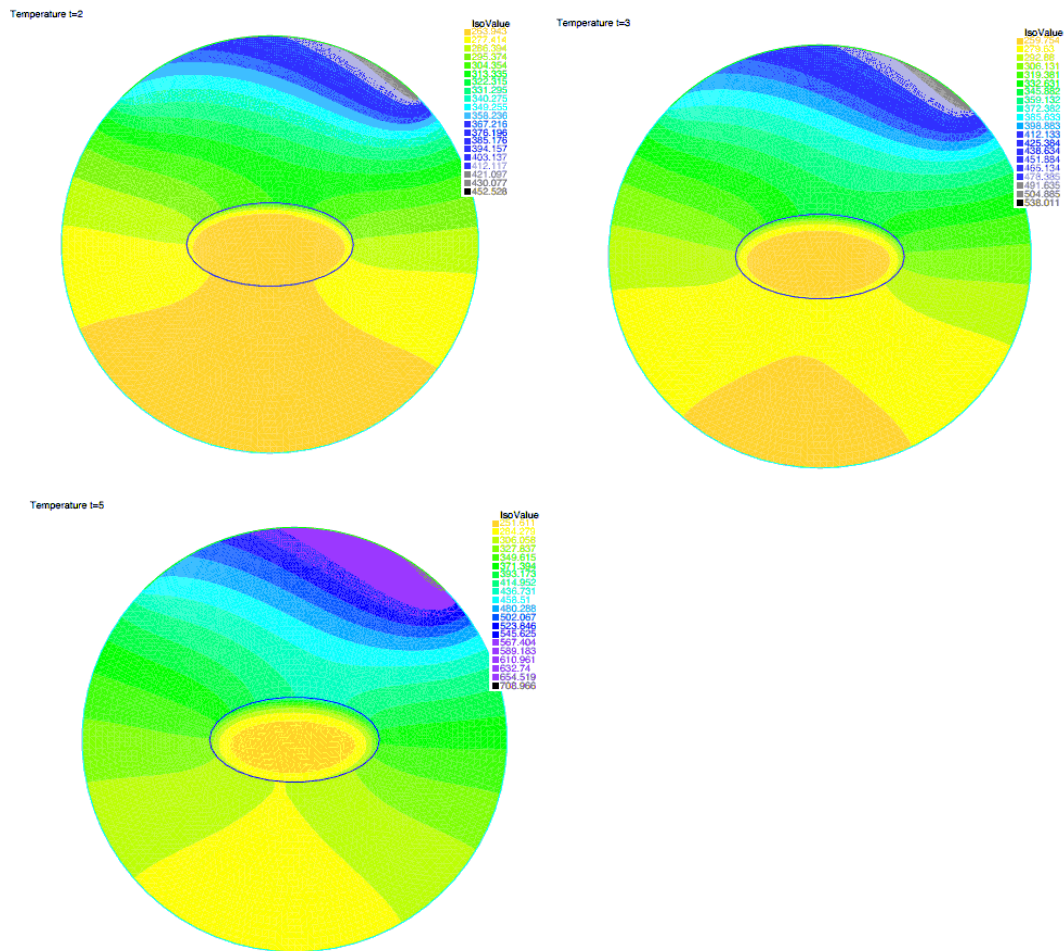
Executing this part into a loop, we can solve the initial example. In the solution , we could see the progress of the changes of heat. It will evolve for different values of t.

The results of this second exercise, we obtain:
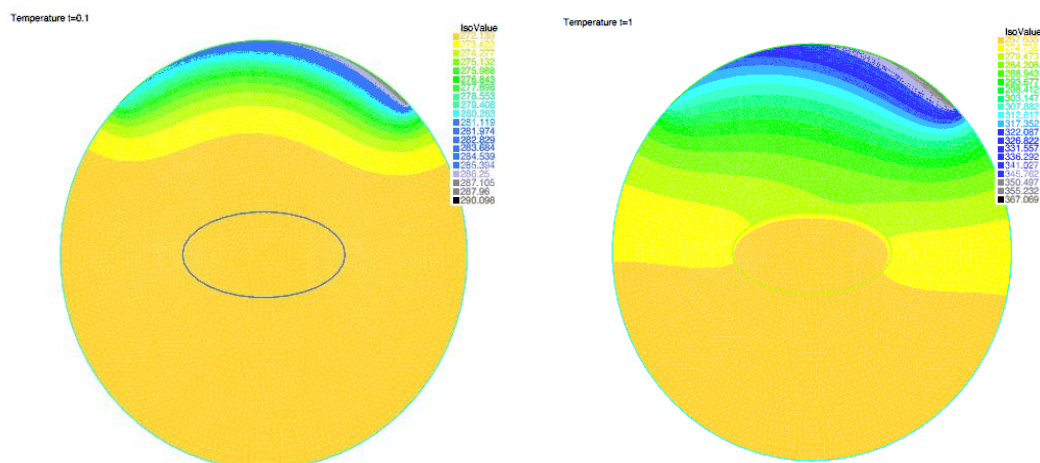
This these graphics obtained by FreeFEM++ we can see the evolution of heat in our domain with the variation on time.

3. Building 3D mesh of the gear

In the following, we proceed to describe the 3D geometric construction of the gear in question via 3D meshes. In particular, the objective is to create a 3D object composed of a finite number of 3D surfaces, which are individual 3D meshes, based on 2D-to-3D transformations.

We begin by building the 3D skin of the gear, which is composed of four separate surfaces (see picture).
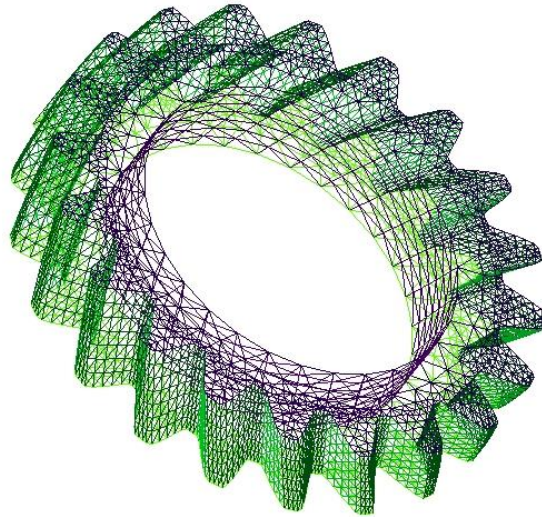
The top and bottom of the gear are flat, annuli-like meshed surfaces, whose outer boundaries are formed using a sinusoidal function, so as to create the appropriate visualization of the teeth of the gear. We further apply a truncation function to it, in order to make the teeth flat, as well as a coordinate transformation, so as to create a shift in them, making the visualization more realistic.
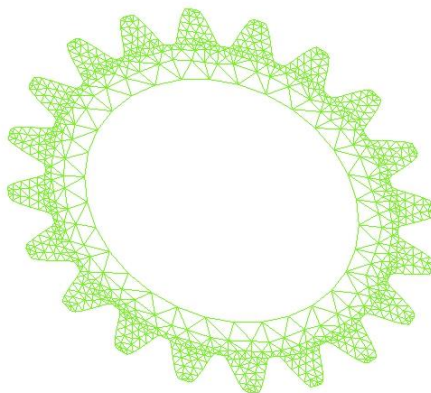
Moreover, we add an inner and outer wall to the object by creating two 2D rectangular meshes, and mapping them into 3D space (introducing an additional coordinate for the height).
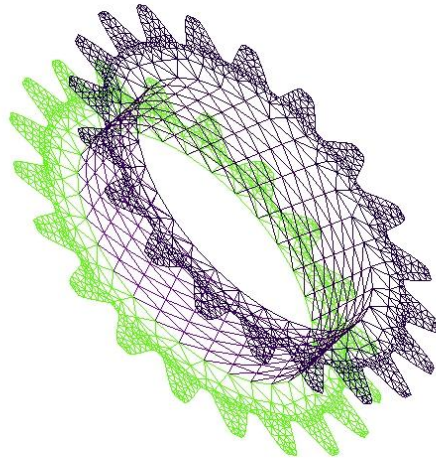

In particular, the outer wall requires the additional application of sinusoidal and truncation functions, as well as another 3D transformation for the shift (similarly as for the top and bottom parts).
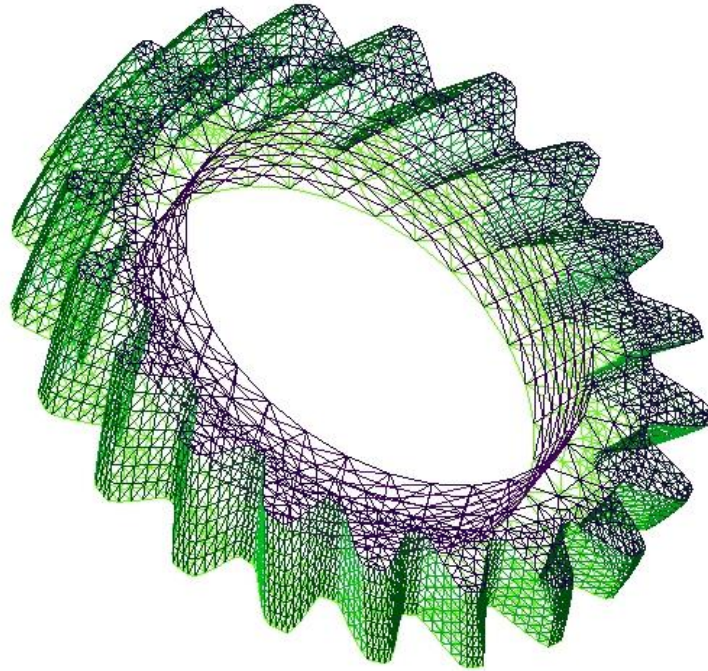
At last, now that we have constructed the skin of the work piece, we proceed to fill it with a mesh of tetrahedra.

All transformations and 2D mesh generations were hereby created via the FreeFEM++ platform using commands such as *mesh3*, *movemesh23* and *movemesh3*. The tetrahedra are automatically generated as well using the command *tetgen*.

We further visualize the obtained object by importing it into the software program *gmsh*, allowing us to make use of effects such as clipping, color variations etc. Another nice addition of this program is that it provides the number of nodes, triangles, tetrahedra etc. of the meshed object in question.
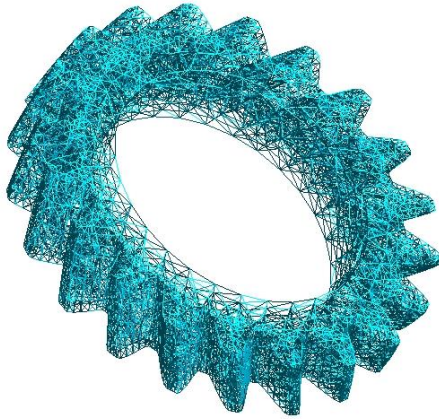
Nodes on Surfaces: 4280
Nodes in Volumes: 1037
# of Triangles: 8560
# of Tetrahedra: 18881

Furthermore, we complete the picture by considering in the work piece in its actual environment, i.e. we create the coil (specific to the electromagnetic heat treatment) around the gear, as well as the box containing them (representing the medium air). The coil is constructed in a similar fashion as the

previous gear skin: we simply a 3D annulus with a small gap in it (determined by the so-called gap angle α which leads to the alternating current, as the composition of inner and outer walls and top and bottom 2D surfaces (from rectangular 2D meshes).

Finally, the box simply consists of four square 2D meshes.
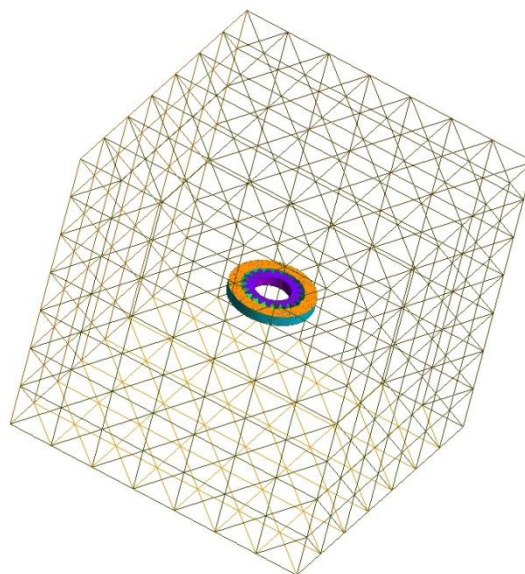
4. Heating and cooling problems

Once we have meshed the gear we can star solving the heating and cooling problems that allows us to transform to Martesite the exterior of the gear and but, at the same time, maintain a ductile interior.

First, we need to heat the work piece at a high temperature range in order to obtain Austenite. There are two ways of heating: induction heating and flame heating. We are going to study the first of these two problems submitting our gear to an alternate current of about 900 Hz for 5.5 seconds.
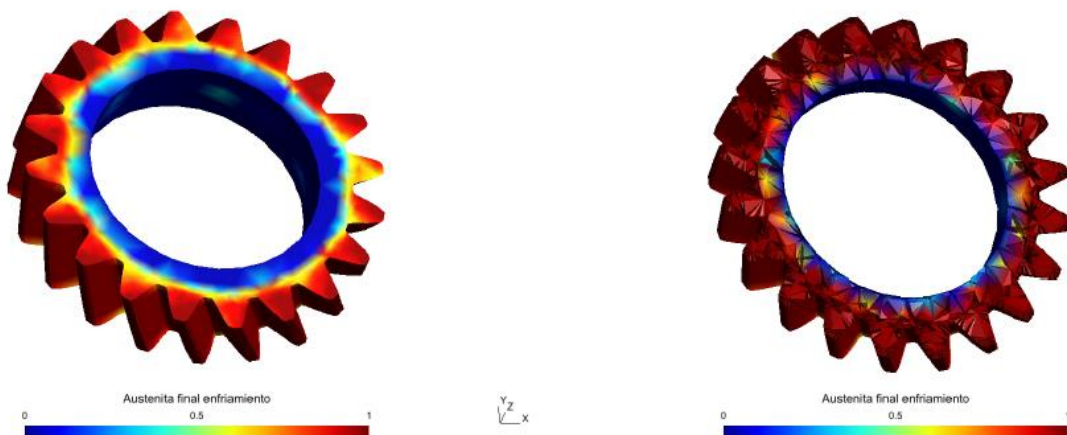
Heating Problem

$$\begin{cases} \rho c \theta_t - \nabla \cdot [k \nabla \theta] = \mathscr{J} - \rho L_1 a_t & in \; \Omega \times (0, T_c), \\[2mm] k \dfrac{\partial \theta}{\partial n} = 0 & on \; \partial \Omega \times (0, T_c), \\[2mm] \theta|_{t=0} = \theta_0 & in \; \Omega, \\[2mm] a_t = f_{1a}(\theta, a) & in \; \Omega \times (0, T_c), \\[2mm] a(x,0) = 0 & in \; \Omega. \end{cases}$$

We can see the result that comes from solving these equations on the Figure below where we can observe that after applying heat on the teeth of the work piece the difference of temperature between the interior and exterior of the gear is considerably high.
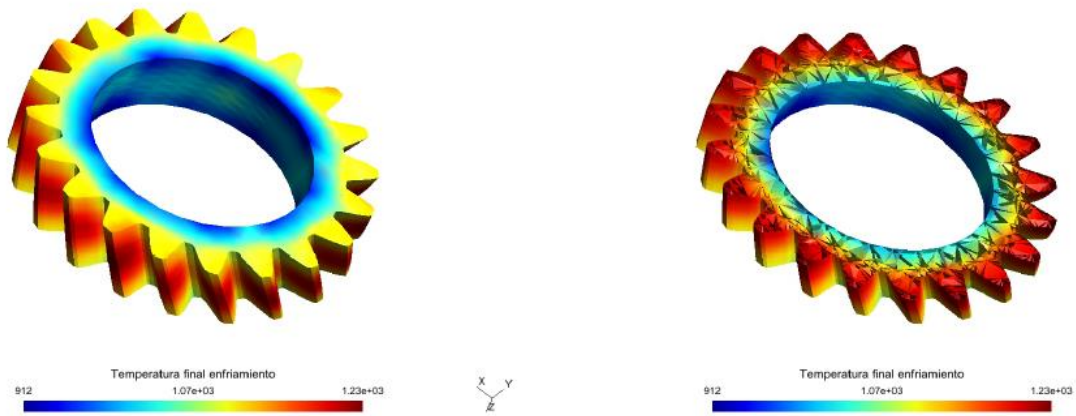
Our aim was to produce Austenite on the exterior so then we can apply a rapid cooling procedure such as *aqua quenching* in order to obtain Martensite. The Figure below shows the result. The red parts correspond to the austenite parts of the gear.


As we mentioned before, in order to transform Austenite into Martensite we need to cool the work piece as fast as possible. The equations below correspond to the cooling problem that needs to be solved with FreeFEM++.

Cooling Problem

$$
\begin{cases}
\rho c \theta_t - \nabla \cdot [k \nabla \theta] = -\rho(L_1 a_t + L_2 m_t) & \Omega \times (0, T_c), \\[2mm]
k \dfrac{\partial \theta}{\partial n} = \alpha(\theta_{ext} - \theta) & on\ \partial\Omega \times (0, T_c), \\[2mm]
\theta|_{t=T_c} = \theta(T_c) & in\ \Omega, \\[2mm]
a_t = f_{2a}(\theta, a, m) & in\ \Omega \times (0, T_c), \\[2mm]
a|_{t=T_c} = a(T_c) & in\ \Omega, \\[2mm]
m_t = f_m(\theta, a, m) & in\ \Omega \times (0, T_c), \\[2mm]
m|_{t=T_c} = 0 & in\ \Omega.
\end{cases}
$$

Let's see the final temperature after 9.5 seconds of cooling.
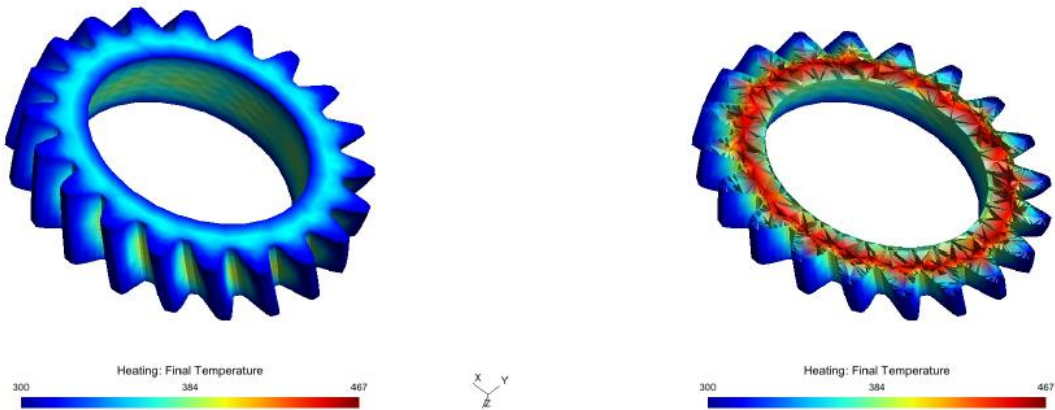
The work piece is still very hot but we have reduced the temperature considerably. Next figure shows that we have transformed almost all the Austenite that remains in the red parts.
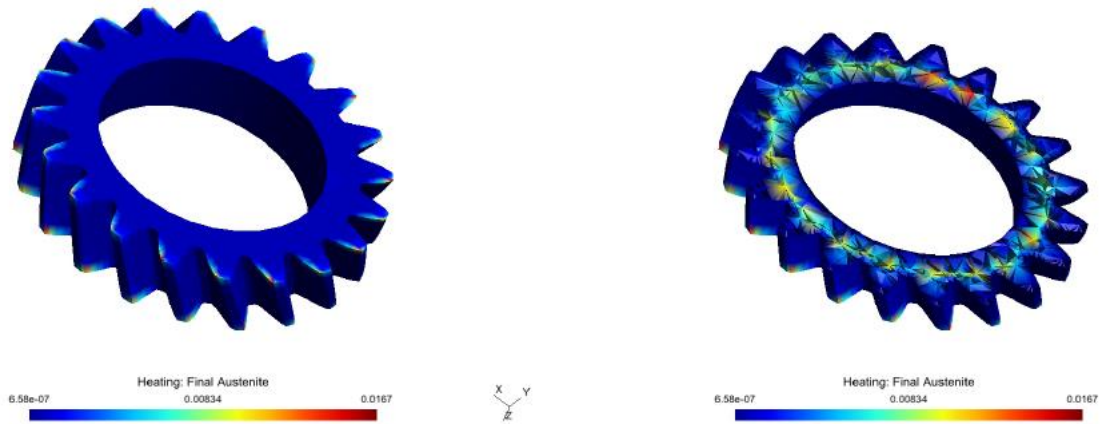
Now we can see the final Martensite exterior wall on the gear. It corresponds to the red parts on the next picture. We can adapt heating and cooling temperatures in order to make this wall thinner.
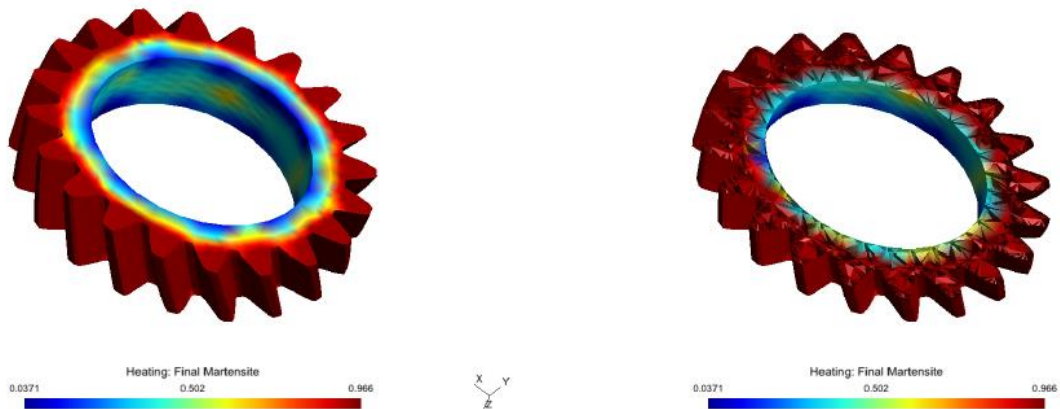
Finally, we show the main difference between induction and flame heating. The last technique allows us to create a thinner exterior wall than the procedure we just describe. We can see this remark on the next picture, where the right gear is the one on which we have applied flame heating.
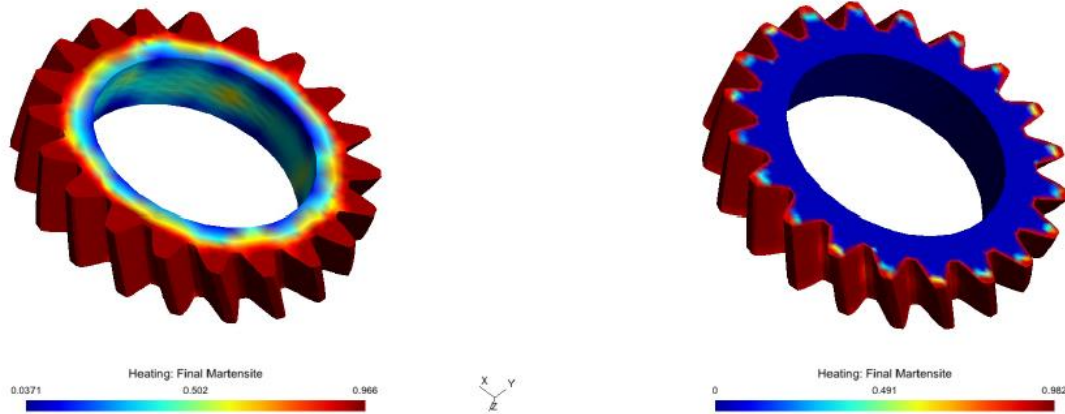
5. Improvements and conclusions

This project enabled us to create a mathematical model for the heat treatment of steel, which describes the temperature distribution, the involved phase transitioning (the occurrence and amount of Martensite and Austenite), as well as how the use of different industrial techniques (flame induced and electromagnetic heat treatment) affects the solution.

Possible extensions of this project could consider the use of more sophisticated numerical techniques such as variable time-stepping for the flame-induced case as well as for the cooling part of the electromagnetic case, as we observed certain irregularities in the temperature distributions for those, probably caused by instabilities of the employed finite difference scheme. One might also add more complexity to the model by considering possible mechanical effects (such as deformation of the material) as well as further electromagnetic effects, such as the change of the electromagnetic term over time. In addition, it would be desirable to consider the model of the entire box of air surrounding the coil and work piece, which so far has been hindered by its very large memory requirement. Finally, a more rigorous convergence analysis might be worth exploring (additionally to exploring variable time-stepping), so as to determine the most appropriate numerical scheme for this model.

All in all, this project allowed us to learn about the material properties of steel, how to use the software FreeFEM++ in 2D and 3D cases, as well as how to construct complex 3D meshed objects modeled on realistic work pieces and visualize them with the help of software such as *gmsh* and *medit*.