UNIVERSIDAD SAN FRANCISCO DE QUITO USFQ

Colegio de Ciencias e Ingeniería

Development of a MatLab Graphical User Interface to Simulate the Thermo-Mechanical Model of Wire + Arc Additive Manufacturing Process.

Proyecto de investigación

Juan Sebastian Palomeque Vargas

Ingeniería Mecánica

Trabajo de titulación presentado como requisito
para la obtención del título de
Ingeniero Mecánico

Quito, 5 de mayo de 2019

UNIVERSIDAD SAN FRANCISCO DE QUITO USFQ

COLEGIO DE CIENCIAS E INGENIERÍA

HOJA DE CALIFICACIÓN
DE TRABAJO DE TITULACIÓN

Development of a MatLab Graphical User Interface to Simulate the Thermo-Mechanical Model of Wire + Arc Additive Manufacturing Process.

Juan Sebastian Palomeque Vargas

Calificación:

Nombre del profesor, Título académico: Edison Bonifaz, Ph.D.

Firma del profesor: ______________________________

Quito, 5 de mayo de 2019

Derechos de Autor

Firma del estudiante: ________________________________

Nombres y apellidos: Juan Sebastian Palomeque Vargas

Código: 00121685

Cédula de Identidad: 0919493197

Lugar y fecha: Quito, 5 de mayo de 2019

Dedicatoria

A mis padres Rommel y María Elena,

Por haberme dado todo su cariño, formar la persona que soy y darme el apoyo necesario durante toda mi vida para seguir el mejor camino. Camino el cual me ha llevado hasta este momento de mi graduación. Han sido mi gran motivación y motor de mi vida para seguir adelante a pesar de las dificultades que se han presentado, pero bajo su bendición las he sabido afrontar. Por esto, les dedico a ellos este logro.

RESUMEN

En el siguiente trabajo, una interfaz gráfica de usuario (GUI) fue desarrollada en MatLab para optimizar y facilitar las simulaciones termo-mecánicas del proceso de manufactura aditiva que fueran desarrolladas en investigaciones previas mediante el código de elementos finitos ABAQUS. La versatilidad del programa permite, asimismo, capturar y graficar los esfuerzos residuales presentes en cada capa que el usuario desee simular. Por otro lado, haciendo uso del programa, se realizaron diferentes simulaciones variando el factor de distribución de calor C, con el fin de determinar su influencia en el proceso de deposición multicapa. Se encontró que los esfuerzos residuales incrementan no solo con el incremento en la velocidad de soldadura, sino también con el incremento del parámetro de distribución de calor. Por otro lado, las deformaciones plásticas incrementan conforme a la velocidad de avance de la soldadura, pero no necesariamente con el aumento del parámetro de distribución de calor C.

Palabras clave: ABAQUS, MatLab, interfaz gráfica de usuario (GUI), manufactura aditiva por arco eléctrico (WAAM) , ciclos térmicos, gradientes térmicos, mecánica de soldadura computacional, modelo termo-mecánico, esfuerzos residuales, distorsión, deformación plástica.

ABSTRACT

In this work, a graphical user interface (GUI) was developed with the software MatLab to facilitate and expedite the study of the thermo-mechanical models of the Wire+Arc Additive Manufacturing (WAAM) that were previously developed in published researches using the finite element analysis code ABAQUS. The versatility of the program allows to capture and plot the residual Mises Stresses developed within the deposited layers that the user wants to simulate. On the other hand, using this WAAM Matlab GUI, different simulations were performed in which the heat flux distribution factor was changed to determine its influence during the multi-layer deposition. It was found that the thermal residual stresses increase with either the increase of weld travel speed or the increase of the heal distribution factor. On the other hand, local plastic strains increase with the increase of welding speed, but not necessarily with the increase of the heat distribution parameter.

Keywords: ABAQUS, MatLab, graphical user interface (GUI), wire + arc additive manufacturing (WAAM), thermal cycles, thermal gradients, computational welding mechanics, thermo-mechanical model, residual stresses, distortion, plastic deformation.

TABLE OF CONTENTS

Introduction **10**

Methodology **13**

- **1. Thermal Model** **13**
- **2. Mechanical Model** **16**
- **3. MatLab GUI** **18**
  - a) Directory folder and file name 20
  - b) Materials properties 20
  - c) Welding Process Parameters and Environment Conditions 22
  - d) Model Geometry and mesh generation 24
  - e) MatLab-ABAQUS interaction 29
  - f) Residual Stresses results 31
  - g) Simulation for heat distribution factor C comparison 33

Results and dicussions **37**

Conclusions **44**

Bibliography **45**

Appendix A: Thermal model MatLab code **47**

Appendix B: Mechanical model MatLab code **62**

Appendix C: MatLab linking Abaqus code **74**

LIST OF TABLES

Table 1: Distribution factor C ranges and nodes across the weld layers. 26
Table 2: Properties for Alloy IN-738 and SAE-AISI 1524. 35
Table 3: List of welding conditions used in the WAAM simulation. 36

LIST OF FIGURES

Figure 1: Gaussian distributed heat source during the WAAM process. 15
Figure 2: Wire Arc Additive Manufacturing GUI 19
Figure 3: Model generated with the WAAM MatLab GUI, with 3 layers. 20
Figure 4: Excel input file format for material properties. 21
Figure 5: DFLUX and SFILM subroutine. 23
Figure 6: Heat Transfer model in welding process 24
Figure 7: Nodes generated across the weld layer with distribution factor C=5.63mm 26
Figure 8: Abaqus thermal model input file format. 28
Figure 9: ABAQUS and subroutine file generated by the MatLab program 29
Figure 10: ABAQUS command executed by the MatLab program. 30
Figure 11: WAAM GUI message display (a) simulation in progress (b) simulation successfully finished. 31
Figure 12: Longitudinal path for residual stress WAAM results. 32
Figure 13: Residual Stress plot from a 3 layers WAAM simulation example. 32
Figure 14: Simulation Set-up for the WAAM process 33
Figure 15: Temperature- dependent properties of SAE AISI 1524 and Alloy IN-738, (a) thermal conductivity, (b) Specific heat, (c) Yield Stress for IN-738 and (d) Yield Stress for AISI 1524. 35
Figure 16: WAAM mechanical simulation behavior at different step times (Mises Stress). .37
Figure 17: Selected paths located on layer 3. 38
Figure 18: Mises Stresses calculated along paths (a) in longitudinal path 1 (b) in transversal path 2. 39
Figure 19: Equivalent plastic strains (PEEQ) calculated along path (a) In the longitudinal Path 1 (b) In the transversal Path 2. 41
Figure 20: Mises stresses and equivalent plastic strains (PEEQ) calculated along Path 1 in three different layers of Weld 3 at time step 120 s (a) Mises stresses (b) Equivalent plastic strains (PEEQ). 42
Figure 21: Mises stresses and equivalent plastic strains (PEEQ) calculated along Path 1 in three different layers of Weld 6 at time step 120 s (a) Mises stresses (b) Equivalent plastic strains (PEEQ). 43

INTRODUCTION

The innovation of manufacturing process should be a priority in modern industry, in order to achieve new objectives in terms of improving the final products obtained from these techniques. It is necessary to emphasize that this innovation must go according to the engineering criteria such as reliability of the design, geometrical accuracy, production rate and efficiency of the manufacturing process (Papacharalampopoulos, Stavropoulos, & Stavridis, 2018). Due to the high competitiveness of the manufacturing industry, this has been forced to focus their knowledge in the development of new technologies that meet the requirements of quality, costs, production and flexibility for it to be integrated in the production process to create more new products (Silva, Barbosa, & Carvalho, 2015).

Additive manufacturing (AM) is a technique to build or generate a three dimensional geometry by adding layer over layer of a selected material that can be either a metal or a polymer (Lindgren & Lundbäck, 2018). In this work, the Wire + Arc Additive Manufacturing (WAAM) process is considered, and it refers to the welding process where a metallic wire is used as a deposit material layer by layer to build the 3D geometry. Furthermore, a robotic or a computer numerical control (CNC) system can be coupled to the process in order to produce more complicated, accurate and functional product made entirely of metal. Within the most important advantages of this process is the low capital cost of the welding equipment, complex design replications and high depositions rates that lead us to have a higher mass production capacity. In addition, the material waste is also an issue that can be mitigated by the implementation of the WAAM process, contrary to the traditional methods (i.e. the machining processes) where a lot of waste material is produced from the initial volume of the workpiece. However, the same geometry can be replicated by the AM process with a final product that needs barely machining to achieve the desire geometrical accuracy (Tabernero, Paskual, Álvarez, & Suárez, 2018).

Although, there has been also some limitations and defects related to this manufacturing process. These problems are mainly caused by thermal cycles that the workpiece is subjected to, involving simultaneously the melting of the top layer and re-melting of the layer underneath, that was previously deposited and is in the solidification stage, increasing in this way the possibility to generated residual thermal stresses within the material that results from the non-uniform temperature changes during this process. These residual stresses can exceed the yield strength of the material that is being added, promoting a geometric distortion on the final product that can produce surface cracks, which affects to the functionality of the product by decreasing its corrosion, fracture and fatigue resistance. (Li, Liu, Fang, & Guo, 2018).

Plenty of methods has been developed to determine and evaluate the magnitude of these residual stresses that are causing all this undesirable property degradation. Among these methods we can find the neutron diffraction measurements, ultrasound and X-rays diffraction. These procedures have proven to be very accurate and valuable to determine whether or not this pieces are going to have a good mechanical performance (Brown et al., 2016). Yet, these techniques represent a high capital cost investment to be executed and the sophisticated equipment must be properly calibrated for its accuracy. Due to this fact, other alternatives have been developed to predict and calculate the magnitude of these residual stresses, that depend greatly on the welding parameters such as voltage, electric current and heat source speed that the workpiece will be subjected to.

Within these alternatives, to perform a reliable prediction and analysis of residual stress, are the denominated thermo-mechanical numerical models, also known as computational welding mechanics (CWM).A typical Finite Element Analysis (FEA) simulates the conditions of temperature, and heat input that the material is being exposed to, during the welding process. Several of these simulations have been validated, as shown by the results obtained by Mukherje, Zahng y DebRoy, in which experimental and simulation results were compared, to conclude that the coupled transient heat transfer and mechanical model is able to calculate residual stresses and distortions caused by the temperature field of the process (2017). Therefore, it is important to emphasize that the simulations performed in the ABAQUS finite element code, replicate in a proper way the WAAM process in terms of working conditions. Despite of this, there are still challenges related to the improvement of the mathematical models, such as their efficiency, interface with the user, generated microstructures information and localization of possible defects in the workpiece (Lindgren & Lundbäck, 2018). Hence, the improvement of these aspect in the FEA will significantly decrease the time and cost necessary to develop new products with different welding parameters.

The following research work describes the development of a user-friendly interface, to enter the parameters of voltage, electric current, and heat source speed of the welding process. In the same way, it is necessary to know the properties of the materials to be used (workpiece and wire materials), such as specific heat, thermal conductivity, coefficient of thermal expansion and yield strength. Once these parameters are known, an automatic mesh that satisfies the specifications described above will be generated, and the simulation of the additive manufacturing process will be performed in the ABAQUS code. After the thermo-mechanical simulations have been successfully completed, the user will be able to observe the Mises Stresses generated in the centerline along the layer, and will be presented in a plot to determine whether or not distortion will occur in the geometry of the final product.

To achieve the proposed goal, a friendly MatLab graphical user interface (GUI) was coupled with a thermo-mechanical program developed in the ABAQUS code. The results obtained in several simulations carried out with different welding parameters are presented to determine the evolution of the thermal cycles and residual stresses during the WAAM process.

METHODOLOGY

1. Thermal Model

Arc welding can be considered as the most common and popular manufacturing process in the industry. This thermal process can be simulated in the finite element software of Abaqus once the welding parameters are known by the user as well as the properties of the materials involved. The following governing differential equation can be solved in order to calculate the temperature at any location considering a moving heat source:

$$\frac{\partial}{\partial x}\left(k\frac{\partial T}{\partial x}\right)+\frac{\partial}{\partial y}\left(k\frac{\partial T}{\partial y}\right)+\frac{\partial}{\partial z}\left(k\frac{\partial T}{\partial z}\right)+\dot{Q}=\rho c_p\frac{\partial T}{\partial t} \tag{1}$$

Where k is the thermal conductivity which depends on the temperature of the material, T is the temperature, ρ is the density, c_p is the specific heat at constant pressure, t is time and $\dot{Q}$ is the internal heat source term (Bonifaz, 2018). Within the hypothesis of this model is that this last term $\dot{Q}$ is zero and we also ignored the latent heat.

In order to solve these differential equations is necessary to know the existing physical boundaries conditions. The solutions in this work, depend on this condition in which we considered that the workpiece initial temperature T_o is equal to 20°C. Another condition that is required to solve the equation is the boundary conditions for the heat exchange between the surface of the workpiece and the surrounding where the heat source due to the arc welding is also involved and can be described by se following equation:

$$-k \left.\frac{\partial T}{\partial y}\right|_{top} + q(r) = h_t(T - T_s) + \sigma\varepsilon(T^4 - T_s^4) \tag{2}$$

Where the both, convective and radiative heat transfer are being considered. In equation (2), h_t is the convection heat transfer coefficient at the top of the workpiece, T_s is the surrounding temperature, σ is the Stefan-Boltzman constant $\left(5.66x10^{-8} \frac{W}{m^2K^4}\right)$ (Incropera, Dewitt, Bergman, & Lavine, 2007), ε is the emissivity. In this model we also considered the heat loss due to convection from all the surfaces of the workpiece where the same coefficient was used $h_t = 10 \frac{W}{m^2K}$. But this is not the case for the area that is being heated up by the welding process, and a value of $h_t = 242 \frac{W}{m^2K}$ is used because of the forced convection caused by the flow of the shielding gas of the welding process as reported in previous publications by Bonifaz and Richards (2009). These effects of heat transfer, moving heat source distribution and constant values can be set by the implementation of the ABAQUS user subroutines FILM and DFLUX.

Within this subroutine's codes that ABAQUS uses in order to simulate the process of additive manufacturing, is the heat input equation $q(x, z, t)$ that the working peace is being subjected during the welding process. The following expression for the heat input is important for the thermal model because it takes into account the moving heat source of the torch, and it also represent the Gaussian power density distribution that is being applied to the surface of the workpiece:

$$q(x,z,t) = \frac{3Q}{\pi C^2}\exp\{-3[(z-vt)^2 + x^2]/\,C^2\,\} \tag{3}$$

$$Q = \eta \text{Vi} \tag{4}$$

Where η the efficiency of the welding process, which takes into consideration the heat losses, V is the voltage, i is the current and C is the characteristic heat distribution parameter. It is important to highlight this last parameter, because this will determine the shape of the Gaussian distribution and also the mesh grid of the material during the simulation, this selection of this values is based in experimental measurements of the fusion zone once the cooling process is completed (Bonifaz, 2018). The next figure will illustrate the Gaussian distribution of the heat source that is being applied to the base material in the process:

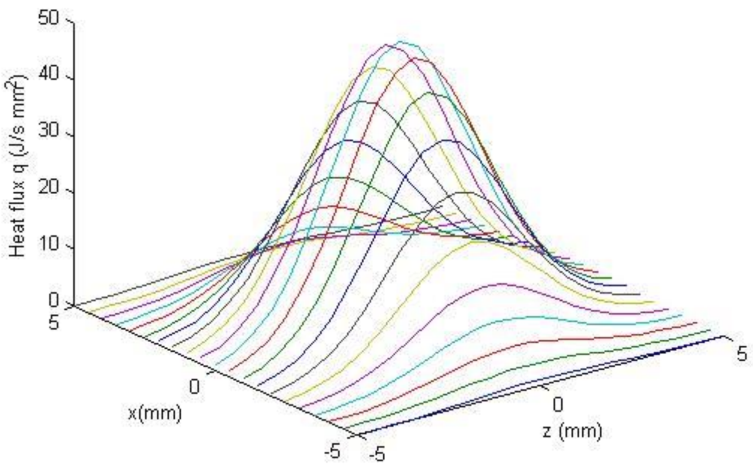


Figure 1: Gaussian distributed heat source during the WAAM process.

2. Mechanical Model

Once the thermal model is done, the next step within this process is the mechanical model of the simulation that is based on the information obtained one the thermal model is simulated. The generalities of the mechanical model reached by Dunne and Petrinic (2005), is explained below:

The total strain rate defined as $\dot{\boldsymbol{\varepsilon}}_{\boldsymbol{ij}}$ is determined by the sum of the elastic $\dot{\boldsymbol{\varepsilon}}_{\boldsymbol{ij}}^{\boldsymbol{e}}$ and plastic $\dot{\boldsymbol{\varepsilon}}_{\boldsymbol{ij}}^{\boldsymbol{e}}$ components of the tensor

$$\dot{\varepsilon}_{ij} = \dot{\varepsilon}_{ij}^{e} + \dot{\varepsilon}_{ij}^{p} \tag{5}$$

The elastic part is defined by the Hooke's Law as:

$$\dot{\varepsilon}_{ij}^{e} = C_{ijkl}\dot{\sigma}_{kl}\ , \ \ C_{ijkl} = \frac{1+v}{E}\ (\delta_{ik}\delta_{jl} - \frac{v}{1+v}\delta_{ij}\,\delta_{kl}) \tag{6}$$

Here, $\boldsymbol{C}_{\boldsymbol{ijkl}}$ is the elastic stiffness matrix of the material, $\dot{\boldsymbol{\sigma}}_{\boldsymbol{kl}}$ is the stress tensor and a dot that is the Newton's notation for derivates with respect to time, $\boldsymbol{E}$ Young's Module, $\boldsymbol{v}$ Poisson's ratio, and $\boldsymbol{\delta}$ is the Kronecker's delta.

This model also contemplates the plastic of the strain that is denoted by the Levy-von Mises equation:

$$\dot{\varepsilon}_{ij}^{p} = \frac{3}{2}\frac{\dot{p}}{q_{vm}}\sigma'_{ij} \tag{7}$$

Where the deviatoric stress tensor is represented by $\boldsymbol{\sigma'_{ij}}$ and the equivalent von Mises plastic strain rate and stress are denoted by the following equations respectively:

$$\dot{p} = \left(\frac{2}{3}\dot{\varepsilon}_{ij}^{p}\dot{\varepsilon}_{ij}^{p}\right)^{1/2} \tag{8}$$

$$q_{vm} = \sqrt{\frac{3}{2}\sigma'_{ij}\,\sigma'_{ij}} \tag{9}$$

This brings us to the next governing equation for the mechanical model

$$\frac{\partial\sigma_{ji}}{\delta x_{j}} = 0 \tag{10}$$

That will be very important for the resolution of the physical problems by means of the finite element methods.

This two models where encoded separately in ABAQUS, but this software has de facility to create a coupled thermos-mechanical model, that takes the information of the transient heat flow analysis, thermal gradients and thermal cycles that are generated in the first step of this process in order to achieve the study of the mechanical model to calculate localized plastic strain and residual stresses (Bonifaz & Richards, 2010). This method can be interpreted as follow:

$$\Delta\varepsilon_{ij} = \Delta\varepsilon_{ij}^{e} + \Delta\varepsilon_{ij}^{p} + \Delta\varepsilon_{ij}^{th} \tag{11}$$

Accounting the classical incremental plasticity theory, the total strain is represented as the sum of the elastic($\boldsymbol{\Delta\varepsilon_{ij}^{e}}$), plastic ($\boldsymbol{\Delta\varepsilon_{ij}^{p}}$), and thermal ($\boldsymbol{\Delta\varepsilon_{ij}^{th}}$) strain increments.

3. MatLab GUI

As explained above, the simulation of the additive manufacturing process consists in the development of two models and a subroutine file. These files must be constantly modified depending on the welding, environmental and geometrical conditions that the user wants to simulate. For this reason, a MatLab program has been developed to facilitate and expedite the study of the process. The user will be able to use this program though a graphical user interface (GUI) denominated WAAM.

No knowledge in ABAQUS modeling or MATLAB programing is necessary to use the proposed GUI interface based on the required parameters (material properties, heat input source, welding parameters, initial temperature condition, heat transfer coefficients and geometrical parameters) showed in Figure 2.

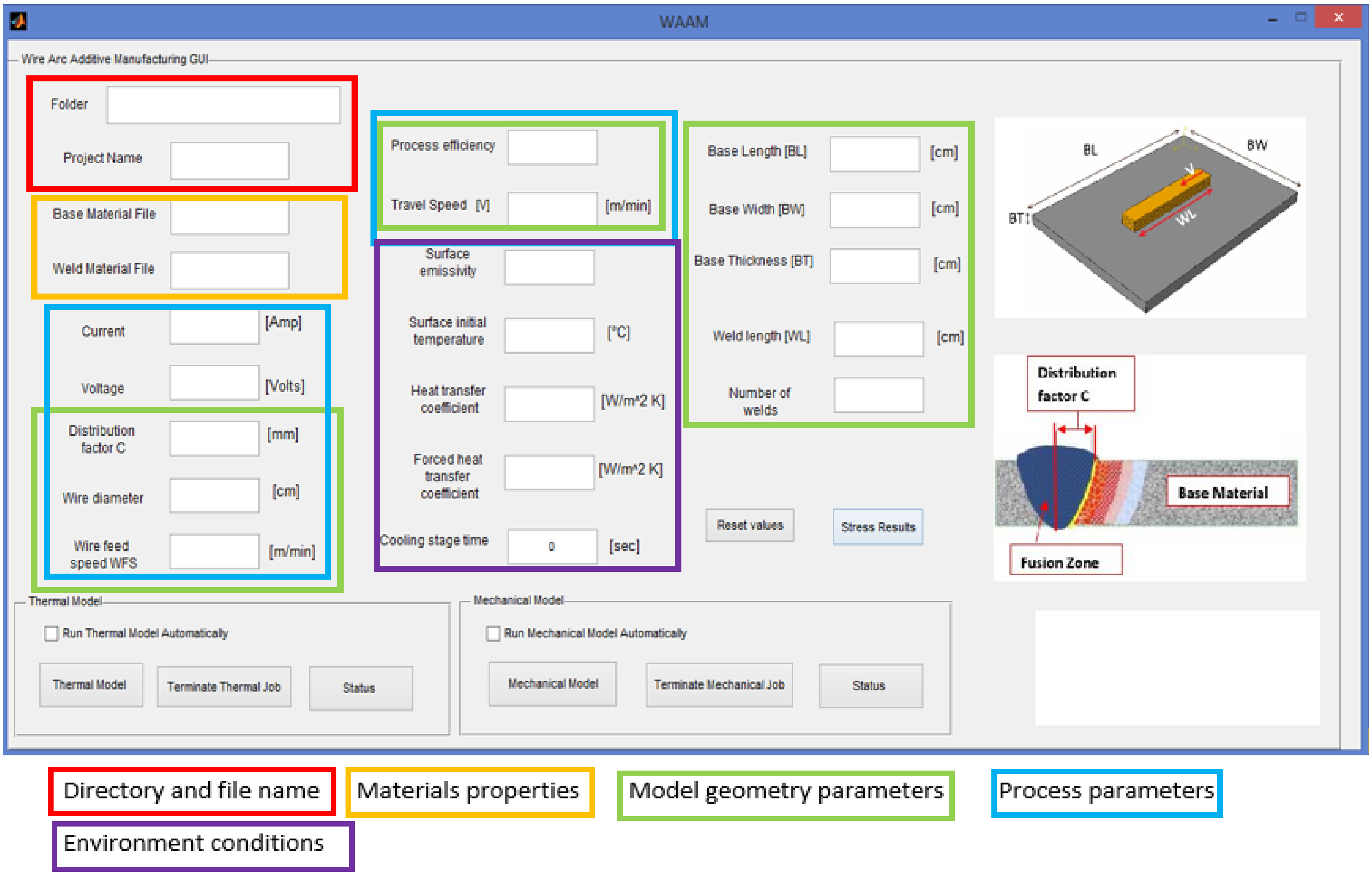


Figure 2: Wire Arc Additive Manufacturing GUI

These inputs, that the user must set up to simulate the process, will define the entire model to build the assembly model shown in Figure 3, where a WAAM model with 3 deposited layers are shown. This means that the MatLab program can be used for any kind of conditions, depending on the information the user has. It is also worth mentioning that the program will work only with 2 types of materials, one for the base metal and the other one that is going to be assigned to all the deposited layers.

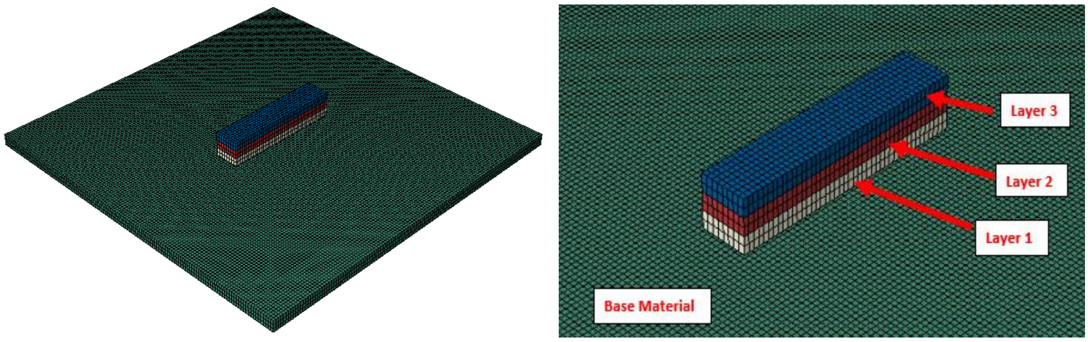


Figure 3: Model generated with the WAAM MatLab GUI, with 3 layers.

a) Directory folder and file name

The first parameter that the user requires to fill out in the MatLab GUI is the directory folder, where all the files needed for this process are going to be created and saved for a more organized work and analysis. Another option that the interface gives to the user is the name of the project, with that said, the program will create the Abaqus input file with extension (.inp) and the Abaqus subroutine (.for) with the name and in the folder selected by the user.

b) Materials properties

According to Figure 2, two Excel files are going to be requested by the program, defined as Base material file and Weld material file. These are going to be 2 different excel files where the material properties are going to be established and saved in the same folder of the chosen directory. Considering that the simulation is going to be a welding process, the temperature depending properties are the most relevant for this analysis, such as the specific heat, density, conductivity, elasticity modulus, poisson ratio, yield stress and the thermal expansion value.

The values of the base material file will define the properties of the base where the weld material will be deposited during the simulation, on the other hand, the weld material file will determine the properties of all the layers added over the base material as illustrated in Figure 3.

All of the mentioned properties have to be stablished in an excel file with a specific format as seen in Figure 4, in order to capture the values correctly and write them into the ABAQUS input file, by using the MatLab command *xlsread()* which stores the values in different columns according to the excel format and then write it into the input file of ABAQUS, justifying in this way, the need of an stablished excel format (MathWorks, 2015) .

Name	SAE AISI 1524 carbon steel

SPECIFIC HEAT	
T[C]	Specific heat [J/KgK]
20	430
100	500
200	550
300	580
400	610
500	650
600	710
700	790
800	865
900	565
1440	630
2500	707
3000	707
5000	707

DENSITY	
T[C]	DENSITY [kg/m^3]
20	7820
1500	7290

CONDUCTIVITY	
T[C]	Conductivity [W/mK]
20	46
100	46
200	45
300	43
400	41
500	38
600	35
700	29
800	24
1500	32
2500	32
5000	32

ELASTIC
Elasticity [Pa]
2E+11

Poission ratio
0.3

PLASTIC	
T[C]	Yield Stress [Pa]
0	3.90E+08
198	2.58E+08
300	2.33E+08
400	2.00E+08
500	1.67E+08
600	1.38E+08
700	9.58E+07
800	7.45E+07
900	4.48E+07
1000	3.06E+07
1100	1.90E+07
1200	9.53E+06
1300	4.76E+06
1400	4.76E+06
1500	4.76E+06

EXPANSION
Thermal Expansion [k^-1]
1.19E-05

Figure 4: Excel input file format for material properties.

c) Welding Process Parameters and Environment Conditions

Once the user has specified the files with the corresponding material properties for the base and the welding material. It is important to know the welding parameters that are: wire diameter, wire feed speed (WFS), current, voltage efficiency, distribution factor C and process efficiency. The MatLab program is going to use these parameters mainly to stablish the heat input source condition based on the distribution factor C, that, as seen before, is going the determine the radius of the heat affected zone caused by the heat input. These parameters along with the travel velocity are going to be considered to write the ABAQUS *DFLUX* and *SFILM* (Figure 5) subroutines which are going to establish the heat source input and heat transfers conditions.

In this case, the DFLUX subroutine is used in ABAQUS FEA to define the heat source distribution with the equation 3, that is in function of position, time and velocity of the source. This condition will be applied in the upper surface of each welded layer, to simulate the material deposition and heat transfer of the welding process. On the other hand, the SFILM allows us to set up the heat transfer coefficient due to natural and forced convection with the environment (Dassault Systèmes Simulia, 2013).

Figure 5: DFLUX and SFILM subroutine.

This subroutine will determine the position of the heat source though the time, and according to its position it will stablish the correct value of the convection heat transfer coefficient for each element depending on whether this element is under the heat source or not, this due to the flow of the shielding gas beneath the nozzle of the torch as shown in Figure 6.

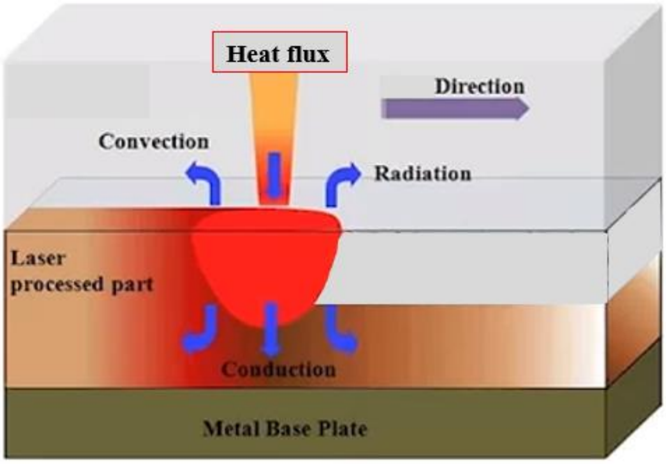


Figure 6: Heat Transfer model in welding process

Other parameters such as the wire diameter, WFS and distribution factor C will also stablish the deposition rate of the material during the process and therefore, the geometry of the welding layers will be defined in conjunction with other parameters explained in the following section.

d) Model Geometry and mesh generation

The user can stablish the geometrical dimensions of the base material but the geometry of the welded layers are defined by the distribution factor C which determines the width of the layer, the length of the layer is provided by the user and the height of the weld is determined by the following equation, based on the material deposition rate and the travel velocity (Linconl Electric, 2006).

$$Deposition\,rate = DR = \frac{\pi D^2}{4} WFS * \eta \ [m^3/s] \tag{12}$$

$$Volume_{deposited} = DR * t = DR * \frac{L_{weld}}{V_{travel}} \ [m^3/s] \tag{13}$$

$$Volume_{weld} = H_{weld} * L_{weld} * W_{weld} \tag{14}$$

$$H_{weld} = \frac{Volume}{L_{weld} * W_{weld}} \ [m^3] \tag{15}$$

Where the deposition rate will depend on the cross-sectional area of the wire with diameter (D), the wire feed speed (WFS) and the efficiency (**η**) of the process which will take into account not only the heat loss but also the material loss due to evaporation during the material deposition. This deposition rate will determine the volume of the material that is being deposited during the simulation and the MatLab program will match the volume of the layer with the volume of the filler metal to find the height of the weld $\boldsymbol{H_{weld}}$.

Once the geometry of the model is stablished, the mesh can be defined for the model. For this, MatLab determines the size of the elements in the welded layers of the model and will use the same size for the base metal. It is worth mentioning that the heat distribution factor C will also have an important role in this part of the model, because it has been stablished the number of nodes across the weld that will be generated according to ranges that are in the MatLab program and reported in the following table.

Table 1: Distribution factor C ranges and nodes across the weld layers.

Ranges (mm)	Nodes	Maximum and minimum distance between nodes
C<3	8	Max.0.375 mm
3≤C<5	10	Max. 0.5mm. Min. 0.3mm
C≥5	12	Min. 0.41mm

As seen in Table 1, there has been selected only pair numbers of nodes, the reason for this is to have a central element across the weld, which will be important in the post-processing stage of the MatLab program from which it is going to get the residual stress magnitudes. It is important to mention that these values where selected in order to have an optimal simulation time with enough information across the weld, the increase of the number of nodes in Table 1 caused the unnecessary increase of the processing time.

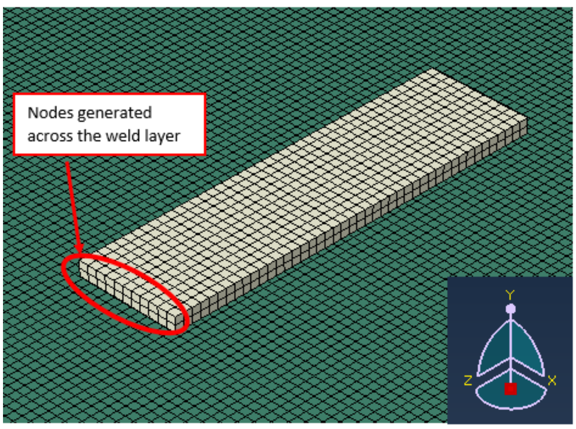
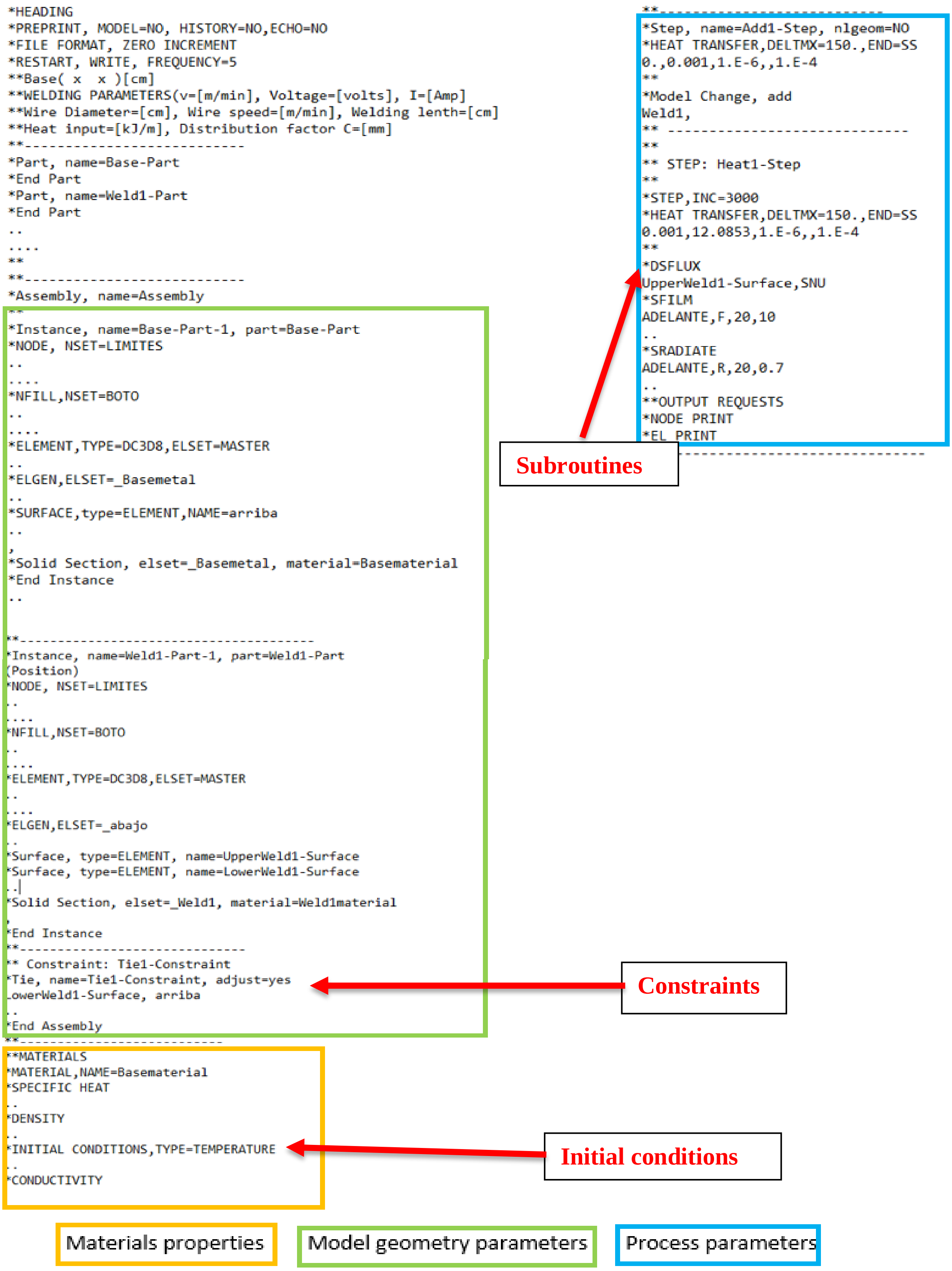


Figure 7: Nodes generated across the weld layer with distribution factor C=5.63mm

These nodes will determine not only the size of the element across the weld (x-direction) but also the element size along the longitudinal path of the weld (z-direction). While the element size in the y-direction will be defined as a half of the weld height $\boldsymbol{H_{weld}}$ defined above. These two layers of elements are necessary to stablish the constrains of the models that will be explained later in this work. It is important to note that the elements generated for the model are defined by ABAQUS as DC3D8 for the thermal model and C3D8 for the mechanical model, which are 3D-8 nodes linear isoperimetric elements (Dassault Systèmes Simulia, 2013). This type element was selected not only because of the geometry of the model, but also for post-processing reasons with the purpose to realize a sub model simulation in the macro and micro scale that will use the information and symmetry of the cubic element to generated these models in an analysis that is not detailed in this work.

The WAAM Matlab GUI is going to use a format similar to the one shown in Figure 8, and it is going to fill the different sections according to the input parameters that the user stablish. It is not necessary for the user to define the constrains, because it will automatically define a *Tie constraint* , involving the surface in contact that are the lower surface of the first weld layer and the upper surface of the base metal for the first step. This constraint will be stablished for both the thermal and the mechanical model, this means that during these simulations, there will be no relative motions between the selected surfaces, depending on how many layers the user requires to analyze (Dassault Systèmes Simulia, 2013).



```
*HEADING
*PREPRINT, MODEL=NO, HISTORY=NO,ECHO=NO
*FILE FORMAT, ZERO INCREMENT
*RESTART, WRITE, FREQUENCY=5
**Base( x  x )[cm]
**WELDING PARAMETERS(v=[m/min], Voltage=[volts], I=[Amp]
**Wire Diameter=[cm], Wire speed=[m/min], Welding lenth=[cm]
**Heat input=[kJ/m], Distribution factor C=[mm]
**--------------------------
*Part, name=Base-Part
*End Part
*Part, name=Weld1-Part
*End Part
..
....
**
**--------------------------
*Assembly, name=Assembly
**
*Instance, name=Base-Part-1, part=Base-Part
*NODE, NSET=LIMITES
..
....
*NFILL,NSET=BOTO
..
....
*ELEMENT,TYPE=DC3D8,ELSET=MASTER
..
*ELGEN,ELSET=_Basemetal
..
*SURFACE,type=ELEMENT,NAME=arriba
..
,
*Solid Section, elset=_Basemetal, material=Basematerial
*End Instance
..

**--------------------------------------
*Instance, name=Weld1-Part-1, part=Weld1-Part
(Position)
*NODE, NSET=LIMITES
..
....
*NFILL,NSET=BOTO
..
....
*ELEMENT,TYPE=DC3D8,ELSET=MASTER
..
....
*ELGEN,ELSET=_abajo
..
*Surface, type=ELEMENT, name=UpperWeld1-Surface
*Surface, type=ELEMENT, name=LowerWeld1-Surface
..|
*Solid Section, elset=_Weld1, material=Weld1material
,
*End Instance
**------------------------------
** Constraint: Tie1-Constraint
*Tie, name=Tie1-Constraint, adjust=yes
LowerWeld1-Surface, arriba
..
*End Assembly
**-------------------------
**MATERIALS
*MATERIAL,NAME=Basematerial
*SPECIFIC HEAT
..
*DENSITY
..
*INITIAL CONDITIONS,TYPE=TEMPERATURE
..
*CONDUCTIVITY

**-------------------------
*Step, name=Add1-Step, nlgeom=NO
*HEAT TRANSFER,DELTMX=150.,END=SS
0.,0.001,1.E-6,,1.E-4
**
*Model Change, add
Weld1,
** ----------------------------
**
** STEP: Heat1-Step
**
*STEP,INC=3000
*HEAT TRANSFER,DELTMX=150.,END=SS
0.001,12.0853,1.E-6,,1.E-4
**
*DSFLUX
UpperWeld1-Surface,SNU
*SFILM
ADELANTE,F,20,10
..
*SRADIATE
ADELANTE,R,20,0.7
..
**OUTPUT REQUESTS
*NODE PRINT
*EL PRINT
------------------------------
```


Figure 8: Abaqus thermal model input file format.

The initial temperature condition is also an option for the user to set up, it can occur that the base metal can be preheated until a certain temperature in order to decrease the temperature gradient during the process to avoid unexpected microstructure within the manufactured pieces. For this reason, the user can determine which initial temperature the base material is going to have for the simulation as shown in Figure 8.

The last section of the Abaqus input file specifies the type of simulation, which is going to be a heat transfer simulation for the first step, in order to get the temperature history for each step of the simulation that will be stored in a file with extension (.fill) with the same name of the project, based on these stored values, a mechanical simulation will be performed. In this section, the subroutines are being called by the file to stablish the heat input source and the heat transfer conditions of the simulation.

e) MatLab-ABAQUS interaction

The MatLab program was developed to run automatically the model generated based on the parameters and the information that the user provides though the graphical user interface. All these parameters will be considered and the WAAM program will generate an ABAQUS file with the extension (.inp) and the subroutine (.for) with the project name that the user chose for the analysis (see Figure 9).

Figure 9: ABAQUS and subroutine file generated by the MatLab program

If it is the case, the user can decide whether the simulation will run automatically once the files are created. This is possible with the implementation of the MatLab command *dos()*, which calls the operating system of Windows to execute the commands specified in the file Mat2Aba.bat that contains the following script:

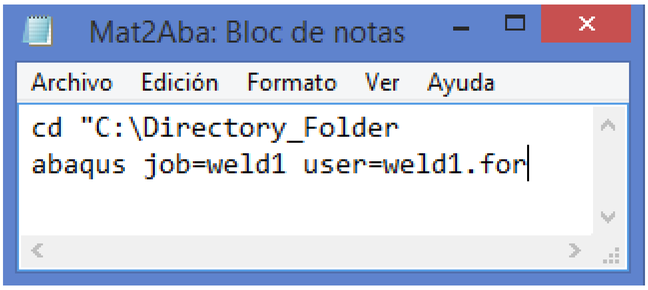
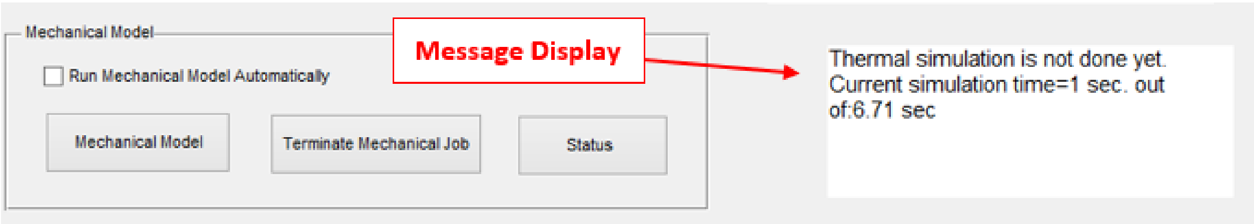


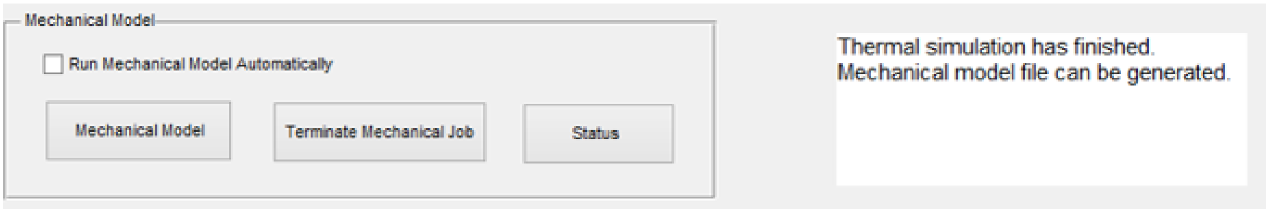
Figure 10: ABAQUS command executed by the MatLab program.

In Figure 10 is possible to see that ABAQUS is going to run the job called weld1 with its subroutine weld1.for which contains the model generated based on the specified welding parameters. Once the thermal simulation is performed and successfully completed, the user can generate the mechanical model which is going to take into consideration the thermal cycles generated within WAAM process. This model can also be simulated automatically by ABAQUS using the same procedure of the first stage of the process, generating a new command which will specify the new file with the extension _mechanical.inp as seen in Figure 9.

Considering that the simulations of this process can take several hours, the user has the option to close the WAAM GUI and let the simulation run by itself. In case that the user wants to inspect the time left to end the simulation successfully, it is possible to open the program and fill only the directory folder and the name of the project, then a message will appear in the program as shown in Figure 11 notifying the current time of the simulation and how much is left to finish it, otherwise, the user will be unable to create or run the mechanical model. This can be made because ABAQUS creates a status file with extension (.sta) that will always be updated while the simulation is running. This status will be compared with the total time necessary stablished based on the travel velocity, length and number of layers.



(a)



(b)

Figure 11: WAAM GUI message display (a) simulation in progress (b) simulation successfully finished.

f) Residual Stresses results

In order to avoid post-processing though the ABAQUS interface, a button called Stress Results has been added to the program which will print a plot of the residual stresses of each layer as seen in

Figure **13**. For this plot, the WAAM GUI will consider the values from a longitudinal Path 1, labeled in Figure 12, at the final step of the simulation, to have the final magnitude of the residual stresses generated within the deposited material.

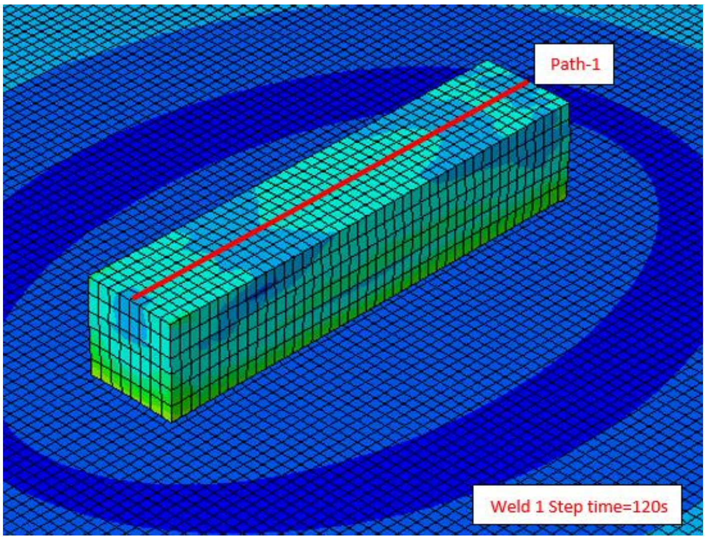


Figure 12: Longitudinal path for residual stress WAAM results.

It is worth mentioning that the plot given by the MatLab program is basic compared to the analysis that can be done directly through the ABAQUS interface where more paths can be chosen and the plastic strains generated in the elements can be plotted as seen later in this work, with that said, it is recommended to use the ABAQUS interface for a more detailed information report of the entire model.

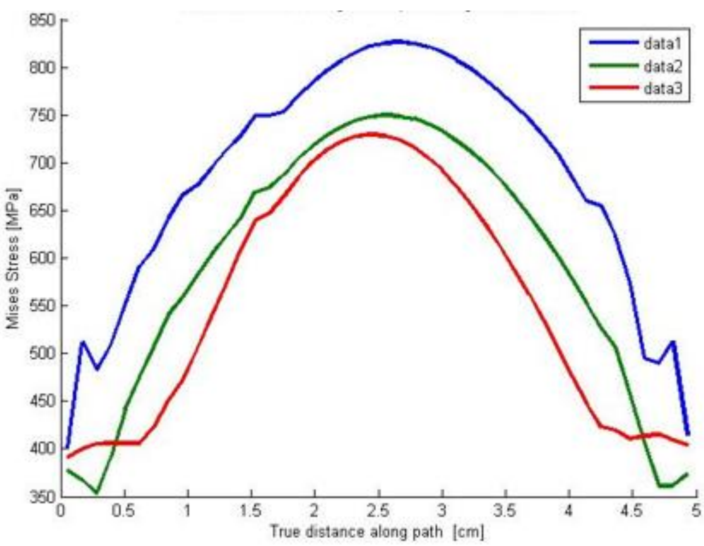


Figure 13: Residual Stress plot from a 3 layers WAAM simulation example.

g) Simulation for heat distribution factor C comparison

For purposes to analyses the effect of the heat distribution factor C, 7 models' simulations have been performed using the WAAM MatLab program. The schematic of the simulations is shown in the following figure:

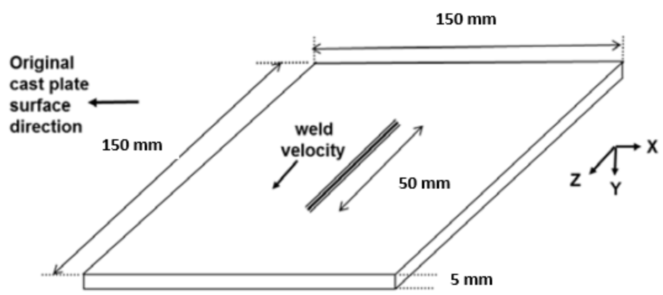
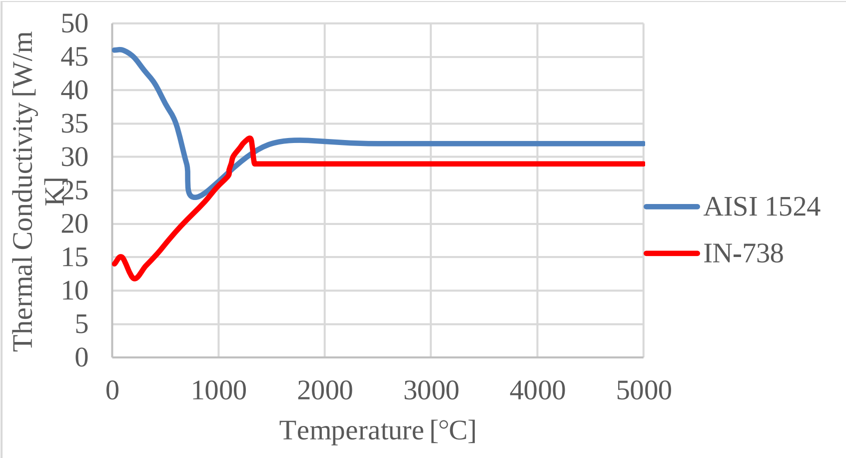
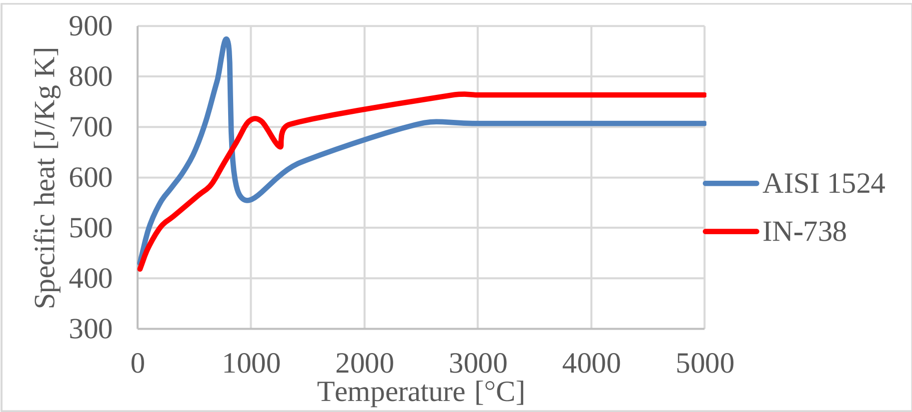


Figure 14: Simulation Set-up for the WAAM process

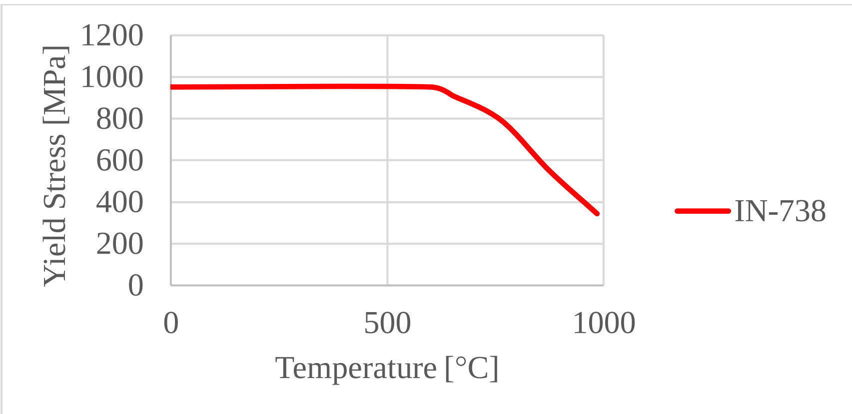
These experiments will consist in the deposition of 3 layers of the Alloy IN-738 over a SAE- AISI 1524 carbon steel plate with the dimensions shown in Figure 14. These temperature-depended properties are obtained with the software J MatPro, which considers the composition of the material to find the approximated property value (Bonifaz & Richards, 2009).


Thermal Conductivity [W/m K]
50
45
40
35
30
25
20
15
10
5
0
0
1000
2000
3000
4000
5000
Temperature [°C]
AISI 1524
IN-738

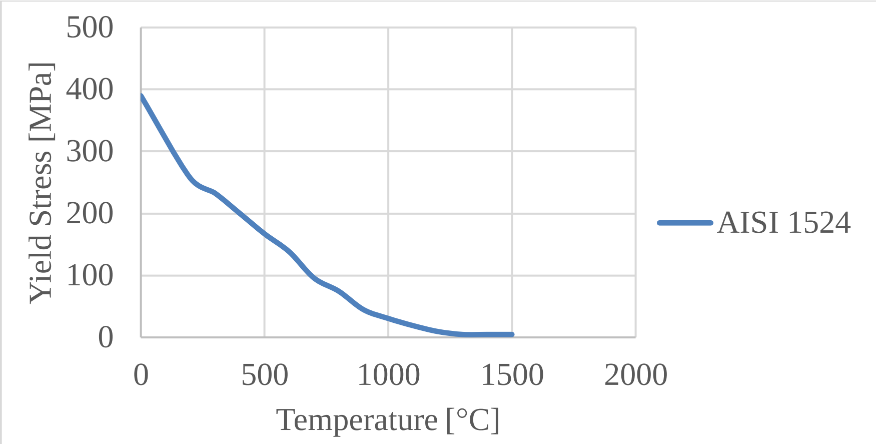
(a)


Specific heat [J/Kg K]
900
800
700
600
500
400
300
0
1000
2000
3000
4000
5000
Temperature [°C]
AISI 1524
IN-738

(b)


Yield Stress [MPa]
1200
1000
800
600
400
200
0
0
500
1000
Temperature [°C]
IN-738

(c)



(d)

Figure 15: Temperature- dependent properties of SAE AISI 1524 and Alloy IN-738, (a) thermal conductivity, (b) Specific heat, (c) Yield Stress for IN-738 and (d) Yield Stress for AISI 1524.

Other constant properties used during the simulation are summarized in the following table.

Table 2: Properties for Alloy IN-738 and SAE-AISI 1524.

Properties	**Alloy IN-738**	**SAE-AISI 1524**
Mean coefficient of thermal expansion	$13.3x10^{-6}\ K^{-1}$	$11.9x10^{-6}\ K^{-1}$
Modulus of elasticity	$204\ GPa$	$200\ GPa$
Density	$8.11\ g/cm^3$	$7.82\ g/cm^3$
Poisson's ratio	0.3	0.3
Surface emissivity	0.7	0.7

For this study, 2 sets of 3 models each, have been determined, among which some of the parameters will remain constant except for the travel speed, voltage and current to maintain a constant heat input in the 2 sets.

Table 3: List of welding conditions used in the WAAM simulation.

	Set 1			**Set 2**		
	Weld 1	**Weld 2**	**Weld 3**	**Weld 4**	**Weld 5**	**Weld 6**
Current (Amp)	200	200	200	208	208	208
Voltage (V)	14.84	20.78	26.72	21.03	29.45	37.86
Travel Velocity (m/min)	0.25	0.35	0.45	0.25	0.35	0.45
Heat input (kJ/m)	712.5	712.5	712.5	1050	1050	1050
Heat distribution parameter *C* (mm)	3.65			5.3		

RESULTS AND DICUSSIONS

To verify that the process was well performed and simulated by the program, it was necessary to analyze the simulation of the (.odb) Abaqus file where the animation of the simulation is stored and can be seen in the following figures.

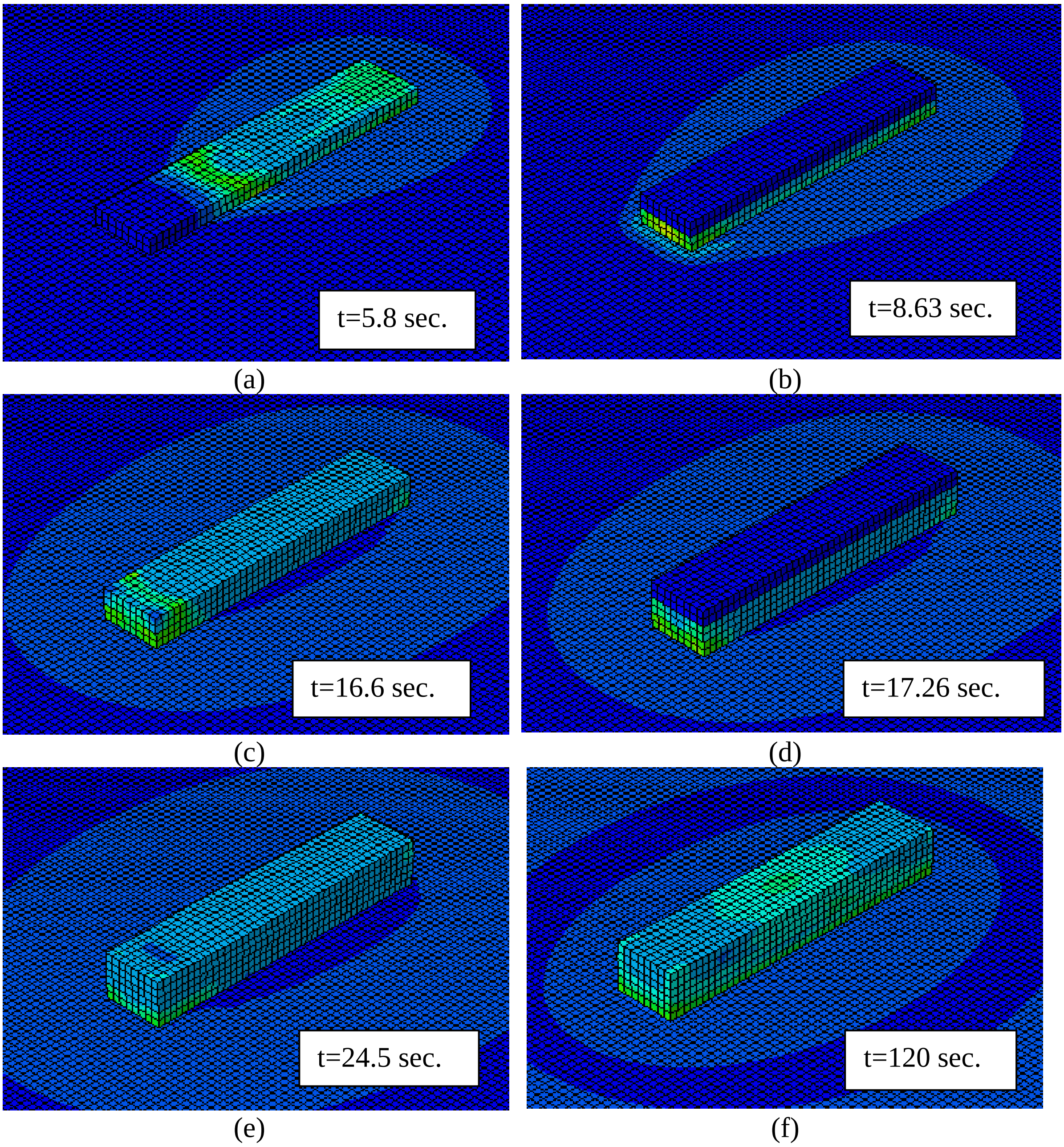


Figure 16: WAAM mechanical simulation behavior at different step times (Mises Stress).

This analysis was made for all the welds reported in this work to ensure that the process has been carried out correctly for all the input parameters that were changing weld by weld. It was important to notice that the heat flux was being applicated to the weld layer for the expected time until the next layer was added to the simulation, and also the distribution size of the heat flux had to have the same size of the weld that was generated based on the input parameters of the user.

Two sets of three welds each, were grouped according to the common heat distribution factor C (see Table 3) which size will be affected by the heat input of the process considering the voltage, current intensity and the travel velocity of the welding process. As explained above, the post-processing will be more detailed using the ABAQUS interface, where different paths where selected to analyze its behavior during the deposition of the material during the WAAM process.

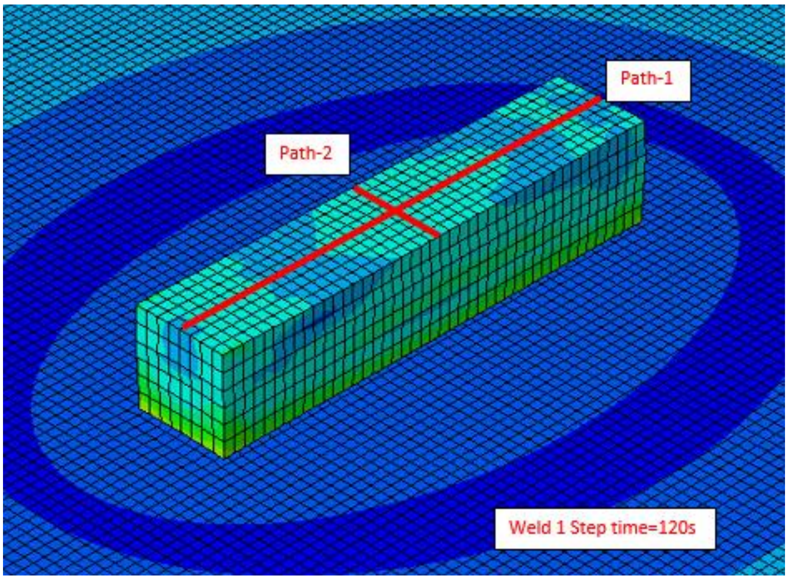
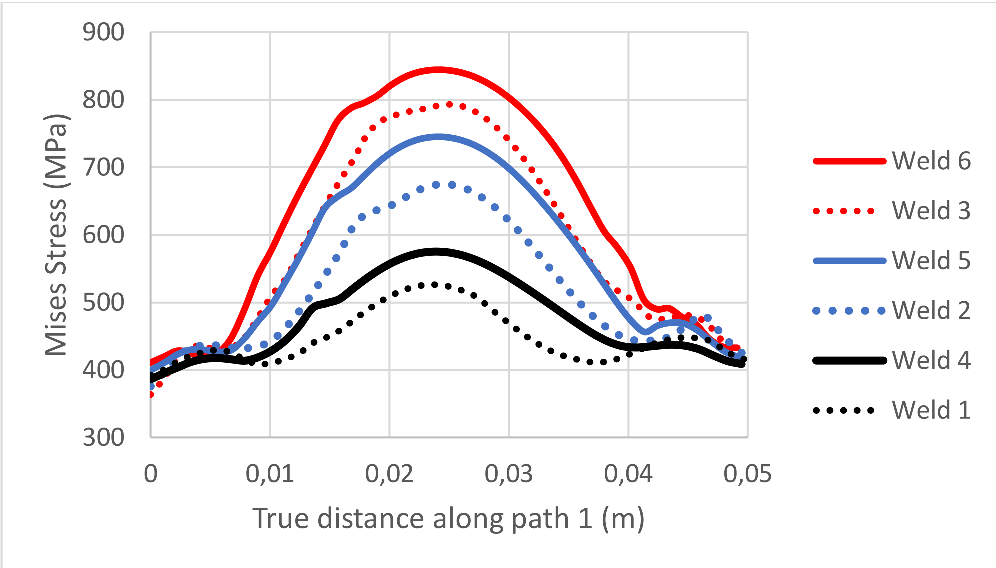
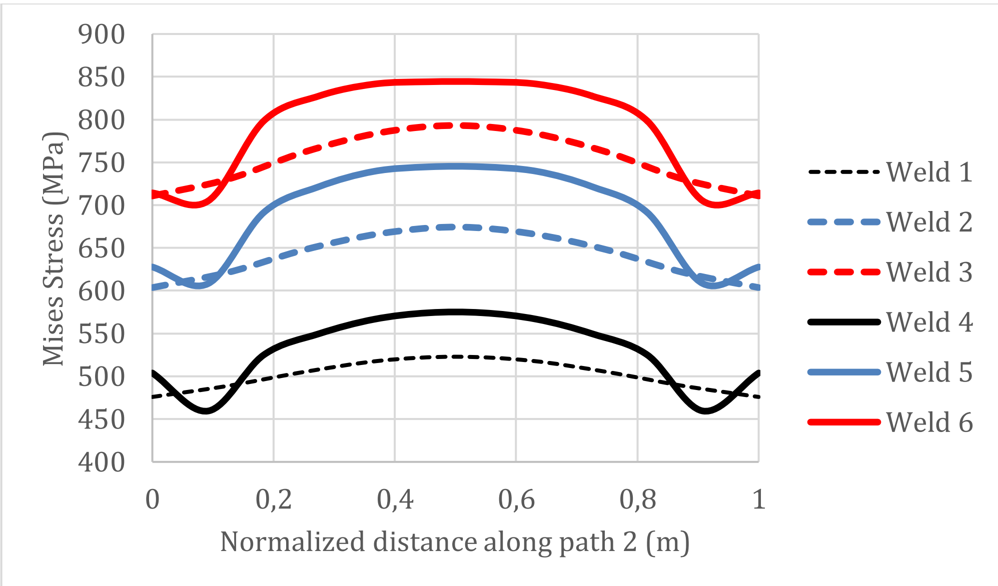


Figure 17: Selected paths located on layer 3.

Figure 18 (a) shows calculated Mises stresses values for the 2 sets of welds, at a time step 120 s. vs. distances along the path 1 of length 50mm, and Figure 18 (b) shows the Mises stresses calculated through the transversal path 2, which length is related to the distribution factor C, for this reason this plot will consider the normalized distance along the path to have a better behavior comparison of the magnitudes of stresses generated across the weld at the last step 120s when the simulation was terminated.



(a)

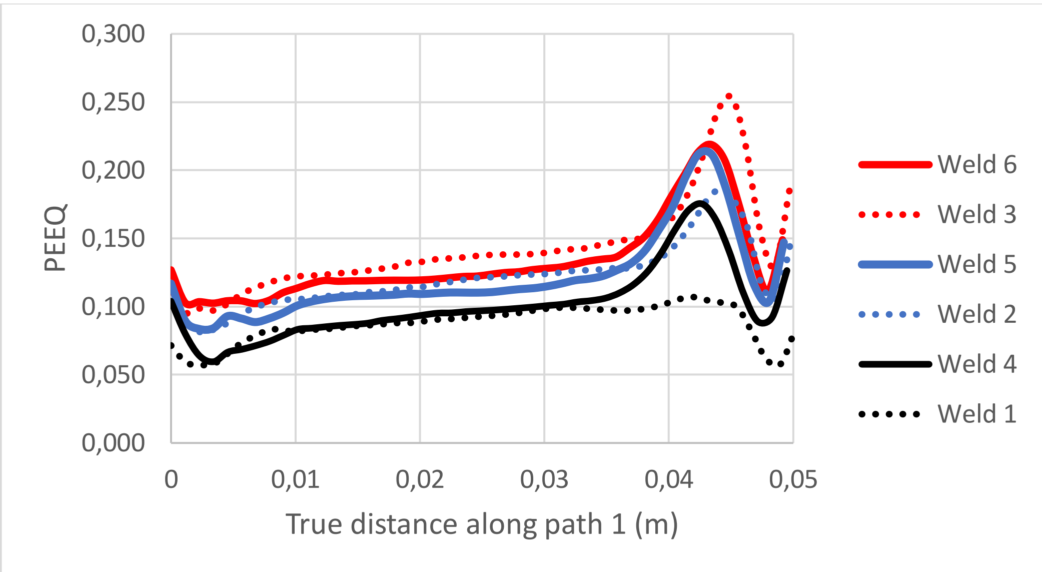


(b)

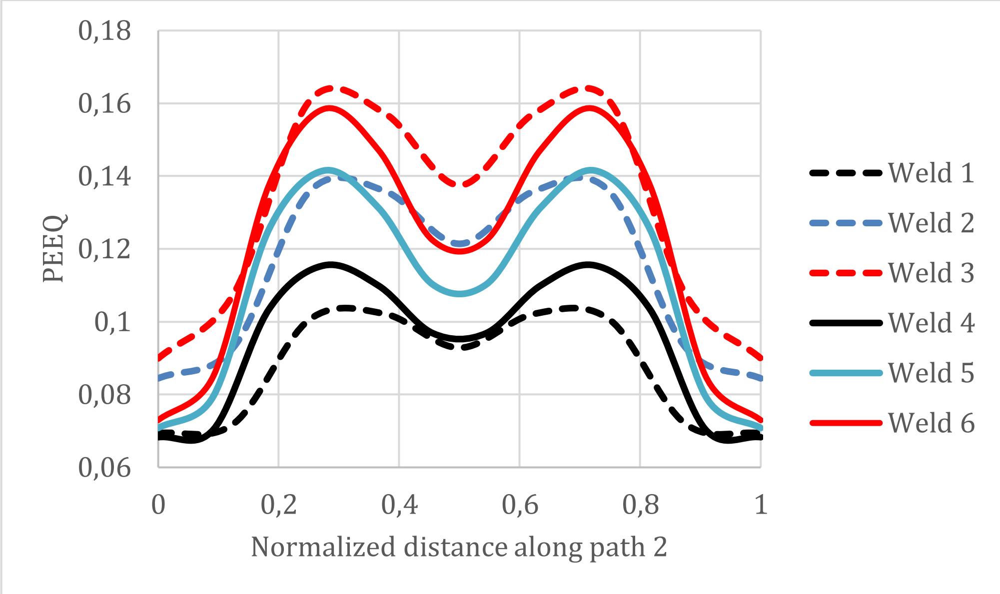
Figure 18: Mises Stresses calculated along paths (a) in longitudinal path 1 (b) in transversal path 2.

It can be clearly seen that the thermal (Mises) stresses increase either with the increase of weld speed *v*, or with the increase of the heat distribution parameter *C*. Note that the heat input was kept constant in each of the two weld sets. It is also observed a concave variation in Mises stresses at the extremes of the transversal path in welds for set 2 (*C*= 5.3 mm). It can be caused by the higher power, bigger *C*, and the effect of convection and radiation heat loss from the weldment (layers) surfaces that act as extended surfaces (fins).

Figure 19 shows a high level of plastic strains calculated along the paths documented in Figure 17. For all the simulations, the highest plastic strains are located near the end of the longitudinal path (see Figure 19 a.). Plastic strains also increase with the increase of welding speed, but not necessarily with the increase of the heat distribution parameter as seen on Figure 19 (a) and (b).



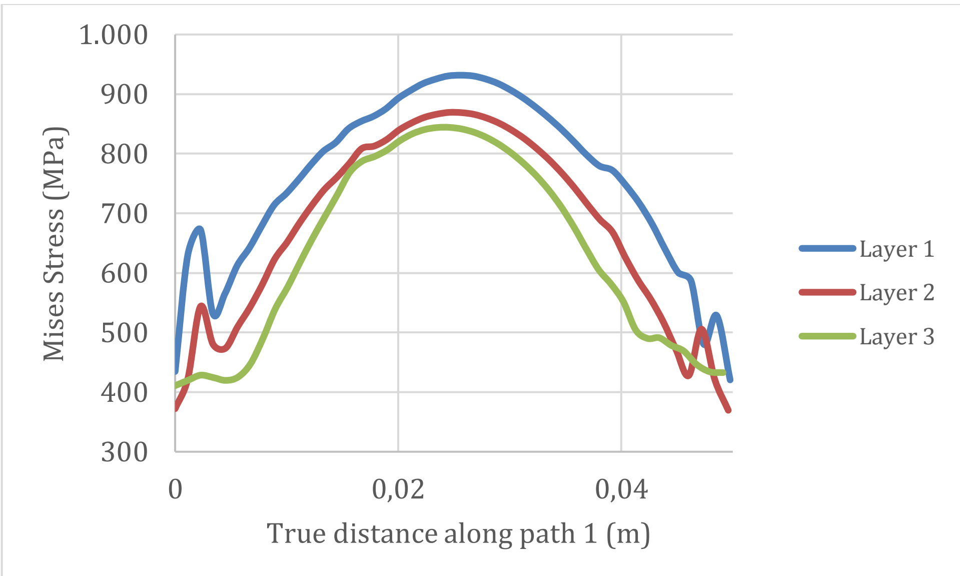
(a)



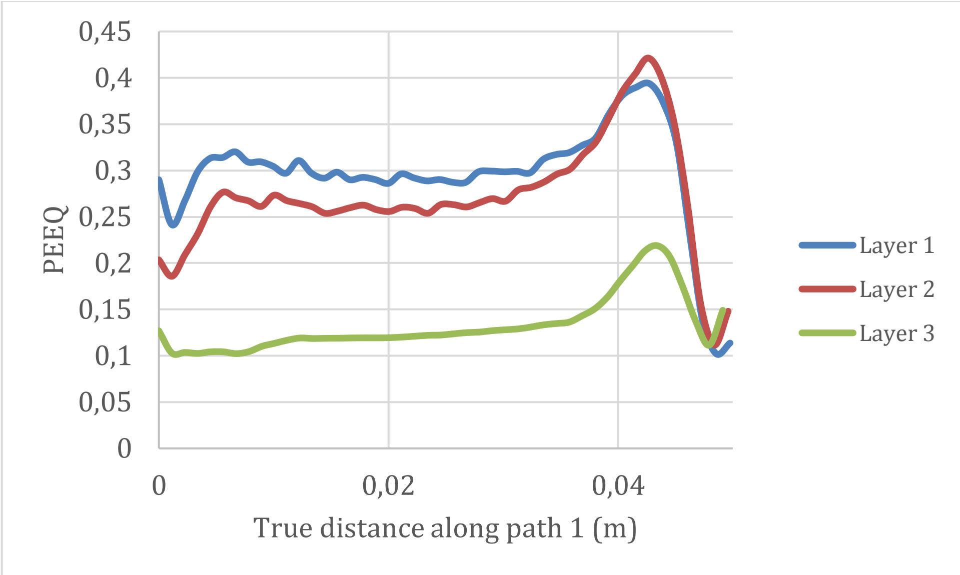
(b)

Figure 19: Equivalent plastic strains (PEEQ) calculated along path (a) In the longitudinal Path 1 (b) In the transversal Path 2.

Figure 20 shows Mises stresses values calculated along Path 1 in three different layers of Weld 3 at time step 120 s. Figure 21 shows Mises stresses values calculated along Path 1 in three different layers of Weld 6 at time step 120 s.



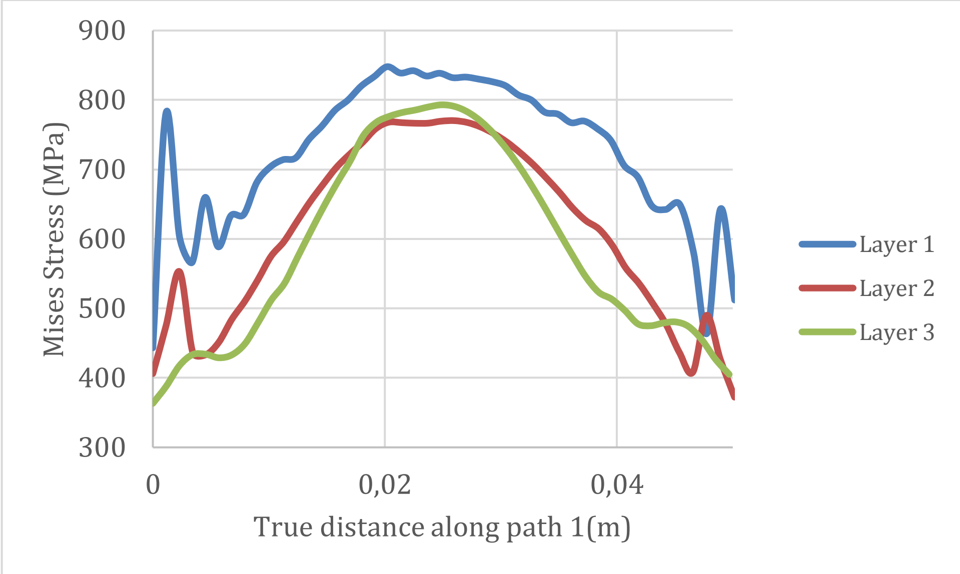
(a)



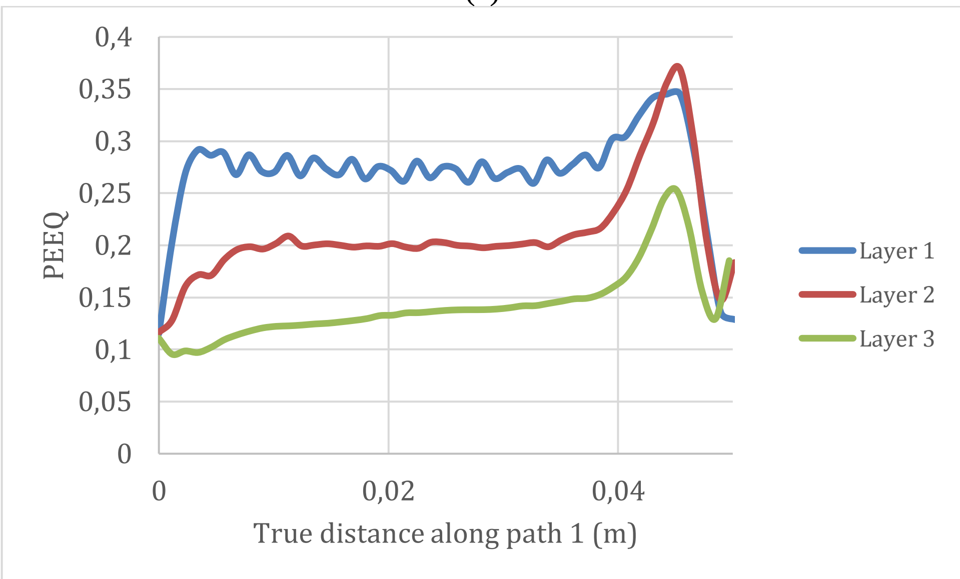
(b)

Figure 20: Mises stresses and equivalent plastic strains (PEEQ) calculated along Path 1 in three different layers of Weld 3 at time step 120 s (a) Mises stresses (b) Equivalent plastic strains (PEEQ).

It is observed in Figure 20 and Figure 21 that the level of Mises stresses and equivalent plastic strains is lower in each new successive AM layer. The order of deposition (Layer 1 →Layer 2 →Layer 3) is shown in Figure 3. As a new layer is deposited over a previous heated one, the relief of thermal stresses and plastic strains occurs by preheating (the more preheated the previous layer, the less the level of thermal stresses and plastic strains in the successive deposited layer). Furthermore, the lowest level of stresses and strains observed in the last deposited AM Layer 3 (Figure 20 and Figure 21), reveals that the relief occurs also by expansion. The top unrestrained weldment surface is free to expand.



(a)



(b)

Figure 21: Mises stresses and equivalent plastic strains (PEEQ) calculated along Path 1 in three different layers of Weld 6 at time step 120 s (a) Mises stresses (b) Equivalent plastic strains (PEEQ).

CONCLUSIONS

As reported in this work, the WAAM MatLab GUI was used to simulate the different conditions that are involved during the WAAM process and to analyze how the heat flux distribution factor affects to the final residual stresses values. The GUI was developed to be used for any possible condition of the process, and also it can be used with different types of material which makes this, a powerful program that can simulate the WAAM process as long as the user knows the parameters and counts with a database of different materials.

It was also recommended that for a better post-processing, the user can use the ABAQUS interface with the (.odb) results file, nevertheless, the developed GUI is able to plot the von Mises Stresses along the weld centerline path that will give an idea of the residual stresses generated within the different layers.

It was found that the thermal (residual) stresses increase with either the increase of weld speed or the increase of the heat distribution parameter. On the other hand, local plastic strains increase with the increase of welding speed, but not necessarily with the increase of the heat distribution parameter.

Similarly, the level of thermal stresses and local plastic strains is lower in each new successive AM layer. As a new layer is deposited over a previously heated one, the relief of thermal stresses and plastic strains occurs by preheating; the more preheated the previous layer, the less the level of thermal stresses and plastic strains in the successive deposited layer. Furthermore, the lowest level of stresses and strains observed in the last deposited AM layer shows that the relief occurs also by expansion because the top unrestrained weldment surface is free to expand.

BIBLIOGRAPHY

Bonifaz, E. A. (2018). Thermo-mechanical analysis in SAE-AISI 1524 carbon steel gas tungsten arc welds. *International Journal Computational Material Science and Surface Engineerign*, *X*. https://doi.org/10.1504/IJCMSSE.2018.095355

Bonifaz, E. A., & Richards, N. L. (2009). Modeling cast IN-738 superalloy gas tungsten arc welds. *Acta Materialia*, *57*(6), 1785–1794. https://doi.org/10.1016/j.actamat.2008.12.022

Bonifaz, E. A., & Richards, N. L. (2010). Stress–Strain Evolution in Cast in-738 Superalloy Single Fusion Welds. *International Journal of Applied Mechanics*, *02*(04), 807–826. https://doi.org/10.1142/S1758825110000767

Brown, D. W., Bernardin, J. D., Carpenter, J. S., Clausen, B., Spernjak, D., & Thompson, J. M. (2016). Neutron diffraction measurements of residual stress in additively manufactured stainless steel. *Materials Science and Engineering A*, *678*(September), 291–298. https://doi.org/10.1016/j.msea.2016.09.086

Dassault Systèmes Simulia. (2013). Abaqus 6.13 Documentation.

Dunne, F., & Petrinic, N. (2005). *Introduction to Computational Plasticity*. New York: Oxford University Press.

Incropera, F., Dewitt, D., Bergman, T., & Lavine, A. (2007). *Fundamentals of Heat and Mass Transfer*. United States of America: John Willey & Sons.

Li, C., Liu, Z. Y., Fang, X. Y., & Guo, Y. B. (2018). Residual Stress in Metal Additive Manufacturing. *Procedia CIRP*, *71*, 348–353. https://doi.org/10.1016/j.procir.2018.05.039

Linconl Electric. (2006). *Gas Metal Arc Welding*. (J. Nadzam, Ed.). Cleveland.

Lindgren, L. E., & Lundbäck, A. (2018). Approaches in computational welding mechanics applied to additive manufacturing: Review and outlook. *Comptes Rendus - Mecanique,* *346*(11), 1033–1042. https://doi.org/10.1016/j.crme.2018.08.004

MathWorks. (2015). Matlab and Simulink R2012b. Retrieved from https://www.mathworks.com

Mukherjee, T., Zhang, W., & DebRoy, T. (2017). An improved prediction of residual stresses and distortion in additive manufacturing. *Computational Materials Science*, *126*, 360–372. https://doi.org/10.1016/j.commatsci.2016.10.003

Papacharalampopoulos, A., Stavropoulos, P., & Stavridis, J. (2018). Adaptive Control of Thermal Processes: Laser Welding and Additive Manufacturing Paradigms. *Procedia CIRP*, *67*, 233–237. https://doi.org/10.1016/j.procir.2017.12.205

Silva, R. J., Barbosa, G. F., & Carvalho, J. (2015). Additive manufacturing of metal parts by welding. *IFAC-PapersOnLine*, *28*(3), 2318–2322. https://doi.org/10.1016/j.ifacol.2015.06.433

Tabernero, I., Paskual, A., Álvarez, P., & Suárez, A. (2018). Study on Arc Welding Processes for High Deposition Rate Additive Manufacturing. *Procedia CIRP*, *68*(April), 358–362. https://doi.org/10.1016/j.procir.2017.12.095

APPENDIX A: THERMAL MODEL MATLAB CODE

```
Function
am=thermal_model(bw,bh,bl,Da,va,e,vp,lw,C,PAS,voltaje,curr,tiempoextra,Name
,sav,BASEMATERIAL,WELDMATERIAL,Tinitial,hconvec,hforced,radiate)
 bw=bw/100; %base width
 bh=bh/100; %bae height
 bl=bl/100; %base length
 Da=Da/100; %wire diameter
 va=va/60; %wire feed speed
 vp=vp/60; %travel speed
 lw=lw/100; %weld length
 C=C/1000; %distribution factor C
t=lw/vp; %time for one layer
DR=(pi*Da^2*va*e)/4; %deposition rate
vol=DR*t;%deposited material volume
anchow=C*2+0.003; %weld width
hw=vol/(lw*anchow);%weld height
filas=2;%layers of elements per weld layer

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%Ranges depending on factor C%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
if C>=0.005
    weldn=12;
end
if C<0.005 && C>=0.003
    weldn=10;
end
if C<0.003
    weldn=8;
end

sizey=hw/filas; %weld element size in y axis
sizex=anchow/weldn; %weld element size in x axis
sizew=sizex; %weld element size in w axis

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%WELD LAYERS GEOMETRY%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%vertex1
AW1=[1,0,0,0];

%vertex2
AW2=weldn;

AW=[AW1;
   AW2,anchow,0,0];

% vertex 3
AW3=weldn*filas+1;

AW=[AW;
    AW3,0,hw,0];

% vertex 4
```

```
AW4=weldn*filas+weldn;

AW=[AW;
   AW4,anchow,hw,0];
% vertex 5
AW5=round((lw/sizew))*AW4+1;
lw=round((lw/sizew))*sizew; %length recalculation according to element size

AW=[AW;
    AW5,0,0,lw];
%vertex6
AW6=AW5+weldn-1;

AW=[AW;
    AW6,anchow, 0,lw];
%vertex7
AW7=AW5+(weldn*filas);

AW=[AW;
    AW7, 0, hw,lw];
%vertex8
AW8=AW5+weldn*filas+weldn-1;

AW=[AW;
    AW8,anchow,hw,lw];

tiempo=lw/vp; %time for one layer
tiempo=roundn(tiempo,-2);
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%BASE GEOMETRY%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%

if bh>=0.01
    brows=6; %layers of element in the base
end
if bh<0.01 && bh>=0.007
    brows=5;
end
if bh<0.007
    brows=3;
end
bnode=round(bw/sizex); %number of nodes across the base

ely=bh/brows; %base element size in y axis
elx=bw/bnode; %base element size in X axis
elz=elx; %base element size in z axis
%vertex1
A1=[1,0,0,0];

%vertex2
A2=bnode;

A=[A1;
   A2,bw,0,0];
%vertex3
A3=bnode*brows+1;

A=[A;
```

```
    A3,0,bh,0];
%vertex4
A4=bnode*brows+bnode;

A=[A;
   A4,bw,bh,0];

%vertex5
A5=round((bl/elz))*A4+1;
bl=round((bl/elz))*elz;
A=[A;
    A5,0,0,bl];

%vertex6
A6=A5+bnode-1;

A=[A;
    A6,bw, 0,bl];
%vertex7
A7=A5+(bnode*brows);

A=[A;
    A7, 0, bh,bl];
%vertex8
A8=A5+bnode*brows+bnode-1  ;

A=[A;
    A8,bw,bh,bl];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%EXCEL FILES DATA COLLECTION%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

s1 = BASEMATERIAL; %collect base material data
s2 = '.xlsx';
s3=sav;
s = strcat(s3,'\',s1,s2);
propiedades_base=xlsread(s);
[fila_prop,columna_prop]=size(propiedades_base);

a1=propiedades_base(:,1);
speceficiheat_temp= a1(~isnan(a1));
a2=propiedades_base(:,2);
speceficiheat_value=a2(~isnan(a2));
specificheat_base=[speceficiheat_value,speceficiheat_temp];

a3=propiedades_base(:,4);
density_temp= a3(~isnan(a3));
a4=propiedades_base(:,5);
density_value=a4(~isnan(a4));
density_base=[density_value,density_temp];

a5=propiedades_base(:,7);
conductivity_temp= a5(~isnan(a5));
a6=propiedades_base(:,8);
conductivity_value=a6(~isnan(a6));
conductivity_base=[conductivity_value,conductivity_temp];

a7=propiedades_base(:,10);
```

```
elasticity_value= a7(~isnan(a7));
a8=propiedades_base(:,11);
elasticity_poison= a8(~isnan(a8));
a9=propiedades_base(:,12);
elasticity_temp= a9(~isnan(a9));
elasticity_base=[elasticity_value,elasticity_poison,elasticity_temp];

a10=propiedades_base(:,14);
yieldstress_temp= a10(~isnan(a10));
a11=propiedades_base(:,15);
yieldstress_value= a11(~isnan(a11));
yieldstress_base=[yieldstress_value,yieldstress_temp];

a11=propiedades_base(:,17);
expansion_value= a11(~isnan(a11));
expansion_base=[expansion_value];


s1 = WELDMATERIAL; %collect weld material data
s2 = '.xlsx';
s3=sav;
s = strcat(s3,'\',s1,s2);
propiedades_suelda=xlsread(s);
[fila_prop,columna_prop]=size(propiedades_suelda);

a1=propiedades_suelda(:,1);
speceficiheat_temp= a1(~isnan(a1));
a2=propiedades_suelda(:,2);
speceficiheat_value=a2(~isnan(a2));
specificheat_weld=[speceficiheat_value,speceficiheat_temp];

a3=propiedades_suelda(:,4);
density_temp= a3(~isnan(a3));
a4=propiedades_suelda(:,5);
density_value=a4(~isnan(a4));
density_weld=[density_value,density_temp];
%
a5=propiedades_suelda(:,7);
conductivity_temp= a5(~isnan(a5));
a6=propiedades_suelda(:,8);
conductivity_value=a6(~isnan(a6));
conductivity_weld=[conductivity_value,conductivity_temp];

a7=propiedades_suelda(:,10);
elasticity_value= a7(~isnan(a7));
a8=propiedades_suelda(:,11);
elasticity_poison= a8(~isnan(a8));
a9=propiedades_suelda(:,12);
elasticity_temp= a9(~isnan(a9));
elasticity_weld=[elasticity_value,elasticity_poison,elasticity_temp];

a10=propiedades_suelda(:,14);
yieldstress_temp= a10(~isnan(a10));
a11=propiedades_suelda(:,15);
yieldstress_value= a11(~isnan(a11));
yieldstress_weld=[yieldstress_value,yieldstress_temp];

a11=propiedades_suelda(:,17);
expansion_value= a11(~isnan(a11));
```

```
expansion_weld=[expansion_value];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%INPUT FILE CREATION FOR ABAQUS%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

s1 = Name;
s2 = '.inp';
s3=sav;
s = strcat(s3,'\',s1,s2);
fi = fopen(s, 'wt');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%FILE HEADING%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

fprintf(fi,'%s\n%s\n%s\n%s\n','*HEADING','*PREPRINT, MODEL=NO,
HISTORY=NO,ECHO=NO','*FILE FORMAT, ZERO INCREMENT','*RESTART, WRITE,
FREQUENCY=5');
fprintf(fi, '%s%g%s%g%s%g%s\n','**Base(',bl*100,' x ',bh*100,' x
',bw*100,')[cm]');
fprintf(fi, '%s%g%s%g%s%g%s\n','**WELDING PARAMETERS(v=',vp*60,'[m/min],
Voltage=',voltaje,'[volts], I=',curr,'[Amp]');
fprintf(fi, '%s%g%s%g%s%g%s\n','**Wire Diameter=',Da*100,'[cm], Wire
speed=',va*60,'[m/min], Welding lenth=',lw*100,'[cm]');
fprintf(fi, '%s%g%s%s%g%s\n','**Heat input=',(10^-
3)*voltaje*curr/vp,'[kJ/m], ','Distribution factor C=',C*1000,'[mm]');
fprintf(fi, '%s\n','**----------------------------');
for k=0:PAS
    if k==0
        fprintf(fi, '%s\n%s\n','*Part, name=Base-Part','*End Part');
    else
        fprintf(fi, '%s%g%s\n%s\n','*Part, name=Weld',k,'-Part','*End
Part');
        fprintf(fi, '%s\n','**');
    end
end
fprintf(fi, '%s\n','**----------------------------');
fprintf(fi, '%s\n','*Assembly, name=Assembly');
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','**----------------------------');
 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%BASE NODES/GEOMETRY%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for j=0:PAS
    if j==0
        fprintf(fi, '%s\n','*Instance, name=Base-Part-1, part=Base-Part');
        fprintf(fi, '%s\n','*NODE, NSET=LIMITES');
        for k=1:length(A)
        fprintf(fi,'%g,%g,%g,%g\n', A(k,1),A(k,2),A(k,3),A(k,4));
        end
        for k=1:8
            fprintf(fi, '%s%g \n %g\n','*NSET,NSET=N',A(k,1),A(k,1));
        end
    fprintf(fi, '%s\n','**----------------------------');
    fprintf(fi, '%s\n
%s%g,%s%g,%g,%g\n','*NFILL,NSET=BOTO','N',A(1,1),'N',A(2,1),bnode-1,1);
    fprintf(fi, '%s\n
%s%g,%s%g,%g,%g\n','*NFILL,NSET=TOPO','N',A(3,1),'N',A(4,1),bnode-1,1);
```

```
    fprintf(fi, '%s\n
%s%g,%s%g,%g,%g\n','*NFILL,NSET=BOTF','N',A(5,1),'N',A(6,1),bnode-1,1);
    fprintf(fi, '%s\n
%s%g,%s%g,%g,%g\n','*NFILL,NSET=TOPF','N',A(7,1),'N',A(8,1),bnode-1,1);

    fprintf(fi, '%s\n','**---------------------------');
    fprintf(fi, '%s\n
%s,%s,%g,%g\n','*NFILL,NSET=BACK','BOTO','TOPO',brows,bnode);
    fprintf(fi, '%s\n
%s,%s,%g,%g\n','*NFILL,NSET=FRONT','BOTF','TOPF',brows,bnode);
    fprintf(fi, '%s\n
%s,%s,%g,%g\n','*NFILL,NSET=_Basemetal','BACK','FRONT',round((bl/elz)),bnod
e*(brows+1));

    fprintf(fi, '%s\n','**---------------------------');

    fprintf(fi, '%s\n
%s%g,%g,%g,%g,%g,%g\n','*ELEMENT,TYPE=DC3D8,ELSET=MASTER','1,1,2,',bnode+2,
bnode+1,A4+1,A4+2,A4+bnode+2,A4+bnode+1);
    fprintf(fi,
'%s\n%s%g,%s%g,%g,%g,%g,%g,%g\n','*ELGEN,ELSET=_Basemetal','1,',bnode-
1,'1,1,',brows,bnode,bnode-1,round((bl/elz)),A4,(bnode-1)*brows);

    fprintf(fi, '%s\n','**');
    fprintf(fi,
'%s\n%g,%g,%g,%g,%g,%g,%g,%g,%g,%g\n%s\n','*ELGEN,ELSET=_abajo',1,bnode-
1,1,1,1,bnode,bnode-1,round((bl/elz)),bnode*(brows+1),(bnode-
1)*brows,'*NSET,NSET=_abajo,ELSET=_abajo');
    fprintf(fi, '%s\n','**');
    fprintf(fi,
'%s\n%g,%g,%g,%g,%g,%g,%g,%g,%g,%g\n%s\n','*ELGEN,ELSET=_arriba',(bnode-
1)*(brows-1)+1,bnode-1,1,1,1,bnode,bnode-
1,round((bl/elz)),bnode*(brows+1),(bnode-
1)*brows,'*NSET,NSET=_arriba,ELSET=_arriba');
    fprintf(fi, '%s\n','**');

    fprintf(fi, '%s\n%s\n','*Solid Section, elset=_Basemetal,
material=BASEMATERIAL',',');
    fprintf(fi, '%s\n','*End Instance');

    fprintf(fi, '%s\n','**---------------------------');

    else
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%WELD NODES/GEOMETRY %%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

        fprintf(fi, '%s%g%s%g%s%g%s\n','*Instance, name=Weld',j,'-Part-
',j,', part=Weld',j,'-Part');
        fprintf(fi,'%g,%g,%g\n',bw/2-anchow/2,bh+(j-1)*hw,bl/2-lw/2);
        fprintf(fi, '%s\n','*NODE, NSET=LIMITES1');

        for k=1:length(AW)
            fprintf(fi,'%g,%g,%g,%g\n', AW(k,1),AW(k,2),AW(k,3),AW(k,4));
        end
        for k=1:8
            fprintf(fi, '%s%g \n %g\n','*NSET,NSET=N',AW(k,1),AW(k,1));
        end
        fprintf(fi, '%s\n','**---------------------------');
```

```
        fprintf(fi, '%s\n
%s%g,%s%g,%g,%g\n','*NFILL,NSET=BOTO','N',AW(1,1),'N',AW(2,1),weldn-1,1);
        fprintf(fi, '%s\n
%s%g,%s%g,%g,%g\n','*NFILL,NSET=TOPO','N',AW(3,1),'N',AW(4,1),weldn-1,1);
        fprintf(fi, '%s\n
%s%g,%s%g,%g,%g\n','*NFILL,NSET=BOTF','N',AW(5,1),'N',AW(6,1),weldn-1,1);
        fprintf(fi, '%s\n
%s%g,%s%g,%g,%g\n','*NFILL,NSET=TOPF','N',AW(7,1),'N',AW(8,1),weldn-1,1);


        fprintf(fi, '%s\n','**---------------------------');
        fprintf(fi, '%s\n
%s,%s,%g,%g\n','*NFILL,NSET=BACK','BOTO','TOPO',filas,weldn);
        fprintf(fi, '%s\n
%s,%s,%g,%g\n','*NFILL,NSET=FRONT','BOTF','TOPF',filas,weldn);
        fprintf(fi, '%s%g\n
%s,%s,%g,%g\n','*NFILL,NSET=_Weld',j,'BACK','FRONT',round((lw/sizew)),weldn
*(filas+1));

        fprintf(fi, '%s\n','**---------------------------');

        fprintf(fi, '%s\n
%s%g,%g,%g,%g,%g,%g\n','*ELEMENT,TYPE=DC3D8,ELSET=MASTER','1,1,2,',weldn+2,
weldn+1,AW4+1,AW4+2,AW4+weldn+2,AW4+weldn+1);
        fprintf(fi,
'%s%g\n%s%g,%s%g,%g,%g,%g,%g,%g\n','*ELGEN,ELSET=_Weld',j,'1,',weldn-
1,'1,1,',filas,weldn,weldn-1,round((lw/sizew)),AW4,(weldn-1)*filas);

        fprintf(fi, '%s\n','**');
        fprintf(fi,
'%s%g\n%g,%g,%g,%g,%g,%g,%g,%g,%g,%g\n%s%g%s%g\n','*ELGEN,ELSET=_LowerWeld'
,j,1,weldn-1,1,1,1,weldn,weldn-1,round((lw/sizew)),weldn*(filas+1),(weldn-
1)*filas,'*NSET,NSET=_LowerWeld',j,',ELSET=_LowerWeld',j);

        fprintf(fi, '%s\n','**');
        fprintf(fi,
'%s%g\n%g,%g,%g,%g,%g,%g,%g,%g,%g,%g\n%s%g%s%g\n','*ELGEN,ELSET=_UpperWeld'
,j,(weldn-1)*(filas-1)+1,weldn-1,1,1,1,weldn,weldn-
1,round((lw/sizew)),weldn*(filas+1),(weldn-
1)*filas,'*NSET,NSET=_UpperWeld',j,',ELSET=_UpperWeld',j);

        fprintf(fi, '%s\n','**');

        fprintf(fi, '%s%g%s\n%s\n','*Solid Section, elset=_Weld',j,',
material=WELDMATERIAL',',');
        fprintf(fi, '%s\n','*End Instance');
    end
    end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%SURFACES CREATION%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%


for k=0:PAS
    if k==0
        fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','**++++++++++++++++++++++++++++++++++++++++++++++');
fprintf(fi, '%s\n','**');
```

```
fprintf(fi,
'%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n','*Nset,nset=Basemetal,in
stance=Base-Part-1','_Basemetal','*Elset,elset=Basemetal,instance=Base-
Part-1','_Basemetal','*Elset,elset=arriba,instance=Base-Part-
1','_arriba','*SURFACE,type=ELEMENT,NAME=arriba','arriba,S5','*Elset,elset=
abajo,instance=Base-Part-
1','_abajo','*SURFACE,type=ELEMENT,NAME=abajo','abajo,S3');
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n%g,%g,%g\n%s\n%s\n%s\n','*ELSET,ELSET=atras,instance=Base-
Part-1,GENERATE',1,(bnode-
1)*brows,1,'*SURFACE,type=ELEMENT,NAME=atras','atras,S1','*NSET,NSET=atras,
ELSET=atras');
fprintf(fi, '%s\n','**');
fprintf(fi,
'%s\n%g,%g,%g\n%s\n%s\n%s\n','*ELSET,ELSET=adelante,instance=Base-Part-
1,GENERATE',round((bl/elz))*((bnode-1)*brows)-(((bnode-1)*brows)-
1),round((bl/elz))*((bnode-
1)*brows),1,'*SURFACE,type=ELEMENT,NAME=adelante','adelante,S2','*NSET,NSET
=adelante,ELSET=adelante');
fprintf(fi, '%s\n','**');
fprintf(fi,
'%s\n%g,%g,%g\n%s\n%s\n%s\n','*ELSET,ELSET=izquierda,instance=Base-Part-
1,GENERATE',1,round((bl/elz))*((bnode-1)*brows)-(bnode-2),bnode-
1,'*SURFACE,type=ELEMENT,NAME=izquierda','izquierda,S6','*NSET,NSET=izquier
da,ELSET=izquierda');
fprintf(fi, '%s\n','**');
fprintf(fi,
'%s\n%g,%g,%g\n%s\n%s\n%s\n','*ELSET,ELSET=derecha,instance=Base-Part-
1,GENERATE',bnode-1,round((bl/elz))*((bnode-1)*brows),bnode-
1,'*SURFACE,type=ELEMENT,NAME=derecha','derecha,S4','*NSET,NSET=derecha,ELS
ET=derecha');
    else
        fprintf(fi, '%s\n','**');
        fprintf(fi,
'%s\n','**++++++++++++++++++++++++++++++++++++++++++++++++');
        fprintf(fi, '%s\n','**');
        fprintf(fi,
'%s%g%s%g%s%g\n%s%g\n%s%g%s%g%s%g\n%s%g\n','*Nset,nset=Weld',k,',instance=W
eld',k,'-Part-',k,'_Weld',k,'*Elset,elset=Weld',k,',instance=Weld',k,'-
Part-',k,'_Weld',k);
        fprintf(fi, '%s\n','**');
        fprintf(fi,
'%s%g%s%g%s%g\n%s%g\n%s%g%s%g%s%g\n%s%g\n%s%g%s\n%s%g%s\n','*Nset,nset=Lowe
rWeld',k,',instance=Weld',k,'-Part-
',k,'_LowerWeld',k,'*Elset,elset=LowerWeld',k,',instance=Weld',k,'-Part-
',k,'_LowerWeld',k,'*Surface, type=ELEMENT, name=LowerWeld',k,'-
Surface','LowerWeld',k,',S3');
        fprintf(fi, '%s\n','**');
        fprintf(fi,
'%s%g%s%g%s%g\n%s%g\n%s%g%s%g%s%g\n%s%g\n%s%g%s\n%s%g%s\n','*Nset,nset=Uppe
rWeld',k,',instance=Weld',k,'-Part-
',k,'_UpperWeld',k,'*Elset,elset=UpperWeld',k,',instance=Weld',k,'-Part-
',k,'_UpperWeld',k,'*Surface, type=ELEMENT, name=UpperWeld',k,'-
Surface','UpperWeld',k',', S5');
        fprintf(fi, '%s\n','**');
        fprintf(fi, '%s\n','**');
        fprintf(fi,
'%s%g%s%g%s%g%s\n%g,%g,%g\n%s%g\n%s%g%s\n%s%g%s%g\n','*ELSET,ELSET=atras',k
,',instance=Weld',k,'-Part-',k,',GENERATE',1,(weldn-
1)*filas,1,'*SURFACE,type=ELEMENT,NAME=atras',k,'atras',k,',S1','*NSET,NSET
=atras',k,',ELSET=atras',k);
```

```
        fprintf(fi, '%s\n','**');
        fprintf(fi, 
'%s%g%s%g%s%g%s\n%g,%g,%g\n%s%g\n%s%g%s\n%s%g%s%g\n','*ELSET,ELSET=adelante
',k,',instance=Weld',k,'-Part-',k,',GENERATE',round((lw/sizew))*((weldn-
1)*filas)-(((weldn-1)*filas)-1),round((lw/sizew))*((weldn-
1)*filas),1,'*SURFACE,type=ELEMENT,NAME=adelante',k,'adelante',k,',S2','*NS
ET,NSET=adelante',k,',ELSET=adelante',k);
        fprintf(fi, '%s\n','**');
        fprintf(fi, 
'%s%g%s%g%s%g%s\n%g,%g,%g\n%s%g\n%s%g%s\n%s%g%s%g\n','*ELSET,ELSET=izquierd
a',k,',instance=Weld',k,'-Part-',k,',GENERATE',1,round((lw/sizew))*((weldn-
1)*filas)-(weldn-2),weldn-
1,'*SURFACE,type=ELEMENT,NAME=izquierda',k,'izquierda',k,',S6','*NSET,NSET=
izquierda',k,',ELSET=izquierda',k);
        fprintf(fi, '%s\n','**');
        fprintf(fi, 
'%s%g%s%g%s%g%s\n%g,%g,%g\n%s%g\n%s%g%s\n%s%g%s%g\n','*ELSET,ELSET=derecha'
,k,',instance=Weld',k,'-Part-',k,',GENERATE',weldn-
1,round((lw/sizew))*((weldn-1)*filas),weldn-
1,'*SURFACE,type=ELEMENT,NAME=derecha',k,'derecha',k,',S4','*NSET,NSET=dere
cha',k,',ELSET=derecha',k);
    end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%MODEL CONSTRAINTS%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n%s\n%s\n','** Constraint: Tie1-Constraint','*Tie, 
name=Tie1-Constraint, adjust=yes','LowerWeld1-Surface, arriba');
fprintf(fi, '%s\n','**');
for k=2:PAS
fprintf(fi, '%s%g%s\n%s%g%s%g%s\n','*Tie, name=Tie',k,'-Constraint, 
adjust=yes','LowerWeld',k,'-Surface,UpperWeld',k-1,'-Surface');
end
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','*End Assembly');


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%MATERIAL PROPERTIES%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

fprintf(fi, '%s\n','**---------------------------');
fprintf(fi, '%s\n','**Materials');
fprintf(fi, '%s\n','**---------------------------');
fprintf(fi, '%s\n','*MATERIAL,NAME=BASEMATERIAL');

fprintf(fi, '%s\n','*SPECIFIC HEAT');
[filaspecific,columnaspecific]=size(specificheat_base);
for k=1:filaspecific
    fprintf(fi, '%g,%g\n',specificheat_base(k,1),specificheat_base(k,2));
end

fprintf(fi, '%s\n','*DENSITY');
[filadensity,columnadensity]=size(density_base);
for k=1:filadensity
    if columnadensity==1
    fprintf(fi, '%g\n',density_base(k,1));
    else
```

```
    fprintf(fi, '%g,%g\n',density_base(k,1),density_base(k,2));
    end
end

fprintf(fi, '%s\n','*CONDUCTIVITY');
[filaconductivity,columnaconductivity]=size(conductivity_base);
for k=1:filaconductivity
    fprintf(fi, '%g,%g\n',conductivity_base(k,1),conductivity_base(k,2));
end
fprintf(fi, '%s\n','**');


fprintf(fi, '%s\n','*MATERIAL,NAME=WELDMATERIAL');

fprintf(fi, '%s\n','*SPECIFIC HEAT');
[filaspecific,columnaspecific]=size(specificheat_weld);
for k=1:filaspecific
    fprintf(fi, '%g,%g\n',specificheat_weld(k,1),specificheat_weld(k,2));
end

fprintf(fi, '%s\n','*DENSITY');
[filadensity,columnadensity]=size(density_weld);
for k=1:filadensity
    if columnadensity==1
    fprintf(fi, '%g\n',density_weld(k,1));
    else
    fprintf(fi, '%g,%g\n',density_weld(k,1),density_weld(k,2));
    end
end

fprintf(fi, '%s\n','*CONDUCTIVITY');
[filaconductivity,columnaconductivity]=size(conductivity_weld);
for k=1:filaconductivity
    fprintf(fi, '%g,%g\n',conductivity_weld(k,1),conductivity_weld(k,2));
end
fprintf(fi, '%s\n','**');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%INITIAL CONDITIONS%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n%s%g\n','*INITIAL
CONDITIONS,TYPE=TEMPERATURE','Basemetal,',Tinitial);

for k=1:PAS
    fprintf(fi, '%s%g,%g\n','Weld',k,Tinitial);
end
fprintf(fi, '%s\n','*PHYSICAL CONSTANTS,ABSOLUTE ZERO=-273.16,STEFAN
BOLTZMANN=5.66E-8');
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','**----------------------------------------------');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%STEPS/SIMULATION CONDITIONS/SUBROUTINES%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fprintf(fi, '%s\n','*Step, name=Remove-Step, nlgeom=NO');
fprintf(fi, '%s\n%s\n','*HEAT TRANSFER,DELTMX=150.,END=SS','0.,0.001,1.E-
6,,1.E-4');
```

```
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','*Model Change, remove');
for k=1:PAS
    fprintf(fi, '%s%g,','Weld',k);
end
fprintf(fi, '\n%s\n','**');
fprintf(fi, '%s\n','**OUTPUT REQUESTS');
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','*Restart, write, frequency=0');
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','**FIELD OUTPUT: F-Output-1');
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n%s\n%s\n','*Output, field, variable=PRESELECT','*Output,
history, frequency=0','*End Step');

fprintf(fi, '%s\n','**-----------------------------------------------');



for j=1:PAS
fprintf(fi, '%s%g%s\n%s\n%s\n%s\n%s\n','*Step, name=Add',j,'-Step,
nlgeom=NO','*HEAT TRANSFER,DELTMX=150.,END=SS','0.,0.001,1.E-6,,1.E-
4','**','*Model Change, add');
fprintf(fi, '%s%g,\n','Weld',j);
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','**OUTPUT REQUESTS');
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','*Restart, write, frequency=0');
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','** FIELD OUTPUT: F-Output-1');
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','*Output, field, variable=PRESELECT');
fprintf(fi, '%s\n','*Output, history, frequency=0','*End Step');
fprintf(fi, '%s\n','** -----------------------------------------------------
------------');
fprintf(fi, '%s\n','**');

fprintf(fi, '%s%g%s\n','** STEP: Heat',j,'-Step');
fprintf(fi, '%s\n','**');
if j==1
    fprintf(fi, '%s\n','*STEP,INC=3000');
else
    fprintf(fi, '%s\n','*STEP,INC=4000');
end
if j==PAS
fprintf(fi, '%s\n%s%g,%s\n','*HEAT
TRANSFER,DELTMX=150.,END=SS','0.001,',tiempo+tiempoextra,'1.E-6,,1.E-4');
else
    fprintf(fi, '%s\n%s%g,%s\n','*HEAT
TRANSFER,DELTMX=150.,END=SS','0.001,',tiempo,'1.E-6,,1.E-4');
end
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','*DSFLUX');
fprintf(fi, '%s%g%s\n','UpperWeld',j,'-Surface,SNU');
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','**');

fprintf(fi, '%s\n','*SFILM');
fprintf(fi,
'%s\n%s%g,%g\n%s%g,%g\n%s%g,%g\n%s%g,%g\n%s%g,%g\n','ARRIBA,FNU','ADELANTE,
```

```
F,',Tinitial,hconvec,'ATRAS,F,',Tinitial,hconvec,'ABAJO,F,',Tinitial,hconve
c,'IZQUIERDA,F,',Tinitial,hconvec,'DERECHA,F,',Tinitial,hconvec);
for k=1:j
    if k==j
        fprintf(fi,
'%s%g%s\n%s%g%s%g,%g\n%s%g%s%g,%g\n%s%g%s%g,%g\n%s%g%s%g,%g\n','UpperWeld',
k,'-
Surface,FNU','ADELANTE',k,',F,',Tinitial,hconvec,'ATRAS',k,',F,',Tinitial,h
convec,'IZQUIERDA',k,',F,',Tinitial,hconvec,'DERECHA',k,',F,',Tinitial,hcon
vec);
    else
        fprintf(fi,
'%s%g%s%g,%g\n%s%g%s%g,%g\n%s%g%s%g,%g\n%s%g%s%g,%g\n%s%g%s%g,%g\n','UpperW
eld',k,'-
Surface,F,',Tinitial,hconvec,'ADELANTE',k,',F,',Tinitial,hconvec,'ATRAS',k,
',F,',Tinitial,hconvec,'IZQUIERDA',k,',F,',Tinitial,hconvec,'DERECHA',k,',F
,',Tinitial,hconvec);
    end
end
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','*SRADIATE');
fprintf(fi,
'%s%g,%g\n%s%g,%g\n%s%g,%g\n%s%g,%g\n%s%g,%g\n','ADELANTE,R,',Tinitial,radi
ate,'ATRAS,R,',Tinitial,radiate,'ARRIBA,R,',Tinitial,radiate,'ABAJO,R,',Tin
itial,radiate,'IZQUIERDA,R,',Tinitial,radiate,'DERECHA,R,',Tinitial,radiate
);
for k=1:j
fprintf(fi,
'%s%g%s%g,%g\n%s%g%s%g,%g\n%s%g%s%g,%g\n%s%g%s%g,%g\n%s%g%s%g,%g\n','UpperW
eld',k,'-
Surface,R,',Tinitial,radiate,'ADELANTE',k,',R,',Tinitial,radiate,'ATRAS',k,
',R,',Tinitial,radiate,'IZQUIERDA',k,',R,',Tinitial,radiate,'DERECHA',k,',R
,',Tinitial,radiate);

end

fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','**OUTPUT REQUESTS');
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','*Restart, write, frequency=0');
fprintf(fi, '%s\n','**');
if j==1
    fprintf(fi, '%s\n%s\n','*NODE PRINT,
NSET=UpperWeld1,FREQ=5,SUMMARY=NO','NT');
    fprintf(fi, '%s\n%s\n','*EL
PRINT,ELSET=UpperWeld1,FREQ=5,SUMMARY=NO,POSITION=CENTROIDAL','TEMP,COORD')
;
else
    fprintf(fi, '%s%g%s\n%s\n','*NODE PRINT,
NSET=LowerWeld',j,',FREQ=5,SUMMARY=NO','NT');
    fprintf(fi, '%s%g%s\n%s\n','*EL
PRINT,ELSET=LowerWeld',j,',FREQ=5,SUMMARY=NO,POSITION=CENTROIDAL','TEMP,COO
RD');
end
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','*Output, field, variable=PRESELECT');
if j==1
```

```
    fprintf(fi, '%s\n%s\n','*ELEMENT
OUTPUT,ELSET=UpperWeld1,POSITION=CENTROIDAL','TEMP');
    fprintf(fi, '%s\n%s\n','*NODE OUTPUT,NSET=UpperWeld1','NT');
    fprintf(fi, '%s\n','**');
    fprintf(fi, '%s\n','*Output, history, frequency=1');
    fprintf(fi, '%s\n%s\n','*ELEMENT OUTPUT,ELSET=UpperWeld1','TEMP');
    fprintf(fi, '%s\n','*NODE OUTPUT,NSET=UpperWeld1','NT');
else
    fprintf(fi, '%s%g%s\n%s\n','*ELEMENT
OUTPUT,ELSET=LowerWeld',j,',POSITION=CENTROIDAL','TEMP');
    fprintf(fi, '%s%g\n%s\n','*NODE OUTPUT,NSET=LowerWeld',j,'NT');
    fprintf(fi, '%s\n','**');
    fprintf(fi, '%s\n','*Output, history, frequency=1');
    fprintf(fi, '%s%g\n%s\n','*ELEMENT OUTPUT,ELSET=LowerWeld',j,'TEMP');
    fprintf(fi, '%s%g\n%s\n','*NODE OUTPUT,NSET=LowerWeld',j,'NT');
end

fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','*NODE FILE','NT');
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','*END STEP');
fprintf(fi, '%s\n','**');

fprintf(fi, '%s\n','** ----------------------------------------------------
------------');
end

fclose(fi);
%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%DFLUX AND SFILM SUBROUTINE %%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
s3 = Name;
s4 = '.for';
s6=sav;
s5= strcat(s6,'\',s3,s4);
fi = fopen(s5, 'wt');
fprintf(fi, '\t%s\n','SUBROUTINE
DFLUX(FLUX,SOL,JSTEP,JINC,TIME,NOEL,NPT,COORDS,JLTYP,');
fprintf(fi, '%s\n','     1 TEMP,PRESS,SNAME)');
fprintf(fi, '%s\n','C');
fprintf(fi, '\t%s\n','INCLUDE ''ABA_PARAM.INC''');
fprintf(fi, '\t%s\n','DIMENSION COORDS(3),FLUX(2),TIME(2)');
fprintf(fi, '\t%s\n','CHARACTER*80 SNAME');
fprintf(fi, '%s\n','C');
fprintf(fi,
'%s\n','C=================================================================
');
fprintf(fi, '\t%s\n','REAL*8
X,Y,Z,Q,LE,ETA,CURR,VOL,V,C,TP,TIEMPO,FLUX,TIME,COORDS');
fprintf(fi,
'%s\n','C=================================================================
');
fprintf(fi, '%s','C');
fprintf(fi, '\t%s\n','LE=0.3E-3');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%WELDING PARAMETERS%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
fprintf(fi,
'\t%s%g\n\t%s%g\n\t%s%g\n\t%s%g\n\t%s%g\n\t%s\n','ETA=',e,'C=',C,'VOL=',vol
taje,'CURR=',curr,'V=',vp,'Q=ETA*VOL*CURR');
fprintf(fi, '\t%s%g%s\n\t%s\n\t%s\n','X=ABS(COORDS(1)-
',bw/2,')','Y=COORDS(2)','Z=COORDS(3)');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%POSITION MAPPING%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fprintf(fi, '\t%s\n\t%s\n\t%s\n\t%s\n\t%s\n\t%s\n','IF(X.LE.C) THEN','GO TO
5','ELSE','FLUX(1)=0.D0','END IF','GO TO 10');
fprintf(fi, '%s%g%s%g%s\n','    5  IF(Z.GE.',bl/2-lw/2,'.AND.Z.LE.',bl/2-
lw/2+lw,')THEN');
fprintf(fi, '\t%s%g%s\n','TP=(COORDS(3)-', bl/2-lw/2,')/V');
fprintf(fi, '\t%s\n','TIEMPO=TP+2*C/V');
fprintf(fi, '\t%s\n','GO TO 8');
fprintf(fi, '\t%s\n','ELSE');
fprintf(fi, '\t%s\n','FLUX(1)=0.D0');
fprintf(fi, '\t%s\n','END IF');
fprintf(fi, '\t%s\n','GO TO 10');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%GAUSSIAN HEAT INPUT%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fprintf(fi, '%s\n','    8    IF(TIME(1).GT.TP.AND.TIME(1).LE.TIEMPO)THEN');
fprintf(fi, '\t%s\n','T=TIME(1)-TP');
fprintf(fi, '\t%s\n','FLUX(1)=((3*Q)/(3.1416*C**2))*EXP(-3*((C-
V*T)**2+X**2)/(C**2))');
fprintf(fi, '\t%s\n','GO TO 10');
fprintf(fi, '\t%s\n','ELSE');
fprintf(fi, '\t%s\n','FLUX(1)=0.D0');
fprintf(fi, '\t%s\n','END IF');
fprintf(fi, '%s\n','C');
fprintf(fi, '%s\n\t%s\n','   10 RETURN','END');

fprintf(fi, '%s\n','C');
fprintf(fi,
'%s\n','C========================================================= ');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%SUBROUTINE SFILM FOR HEAT TRANSFER%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fprintf(fi, '\t%s\n','SUBROUTINE
FILM(H,SINK,TEMP,KSTEP,KINC,TIME,NOEL,NPT,');
fprintf(fi, '\t%s\n','1 COORDS,JLTYP,FIELD,NFIELD,SNAME,NODE,AREA)');
fprintf(fi, '%s\n','C');
fprintf(fi, '\t%s\n','INCLUDE ''ABA_PARAM.INC''');
fprintf(fi, '%s\n','C');
fprintf(fi, '\t%s\n','DIMENSION H(2),TIME(2),COORDS(3), FIELD(NFIELD)');
fprintf(fi, '\t%s\n','CHARACTER*80 SNAME');
fprintf(fi, '%s\n','C');
fprintf(fi,
'%s\n','C==============================================================
');
fprintf(fi, '\t%s\n','REAL*8 X,Y,Z,LE,V,C,TP,TIEMPO,H,TIME,COORDS,TEMP');
fprintf(fi,
'%s\n','C==============================================================
');
fprintf(fi, '%s','C');
fprintf(fi, '\t%s\n','LE=0.3E-3');
```

```
fprintf(fi, '\t%s%g\n','ETA=',e);
for k=1:2
    if k==1
fprintf(fi, '\t%s%g%s\n \t%s%g\n \t%s%g\n \t%s%g\n \t%s%g\n
\t%s\n','if(kstep.LE.',3*k,')then','C=',C,'VOL=',voltaje,'CURR=',curr,'V=',
vp,'go to 2');
    else
fprintf(fi, '\t%s%g%s\n \t%s%g\n \t%s%g\n \t%s%g\n \t%s%g\n \t%s\n','else
if(kstep.LE.',3*k,')then','C=',C,'VOL=',voltaje,'CURR=',curr,'V=',vp,'go to
2');
    end
end
fprintf(fi, '\t%s\n','else');
fprintf(fi,
'\t%s%g\n\t%s%g\n\t%s%g\n\t%s%g\n','C=',C,'VOL=',voltaje,'curr=',curr,'v=',
vp);
fprintf(fi, '%s\n','   2    Q=ETA*VOL*CURR');
fprintf(fi, '\t%s\n','end if');
fprintf(fi, '\t%s%g%s\n\t%s\n\t%s\n','X=ABS(COORDS(1)-
',bw/2,')','Y=COORDS(2)','Z=COORDS(3)');
fprintf(fi, '\t%s\n','IF(X.LE.C) THEN');

fprintf(fi, '\t%s\n','GO TO 5');
fprintf(fi, '\t%s\n','ELSE');
fprintf(fi, '\t%s%g\n','H(1)=',hconvec);
fprintf(fi, '\t%s\n','END IF');
fprintf(fi, '\t%s\n','GO TO 10');
fprintf(fi, '%s%g%s%g%s\n','   5  IF(Z.GE.',bl/2-lw/2,'.AND.Z.LE.',bl/2-
lw/2+lw,')THEN');
fprintf(fi, '\t%s%g%s\n','TP=(COORDS(3)-', bl/2-lw/2,')/V');
fprintf(fi, '\t%s\n','TIEMPO=TP+2*C/V');
fprintf(fi, '\t%s\n','GO TO 8');
fprintf(fi, '\t%s\n','ELSE');
fprintf(fi, '\t%s%g\n','H(1)=',hconvec);    %coeficiente de conveccion
fprintf(fi, '\t%s\n','END IF');
fprintf(fi, '\t%s\n','GO TO 10');
fprintf(fi, '%s\n','   8   IF(TIME(1).GT.TP.AND.TIME(1).LE.TIEMPO)THEN');
fprintf(fi, '\t%s%g\n','H(1)=',hforced);
fprintf(fi, '\t%s\n','GO TO 10');
fprintf(fi, '\t%s\n','ELSE');
fprintf(fi, '\t%s%g\n','H(1)=',hconvec);
fprintf(fi, '\t%s\n','END IF');
fprintf(fi, '%s\n','C');
fprintf(fi, '%s\n\t%s\n','   10 RETURN','END');
fprintf(fi, '%s\n','C');
 fclose(fi);

 end
```

APPENDIX B: MECHANICAL MODEL MATLAB CODE

```
function
i=mechanical_model(bw,bh,bl,Da,va,e,vp,lw,C,PAS,voltaje,curr,tiempoextra,Na
me,sav,BASEMATERIAL,WELDMATERIAL,Tinitial)
bw=bw/100; %base width
 bh=bh/100; %bae height
 bl=bl/100; %base length
 Da=Da/100; %wire diameter
 va=va/60; %wire feed speed
 vp=vp/60; %travel speed
 lw=lw/100; %weld length
 C=C/1000; %distribution factor C

t=lw/vp; %time for one layer
DR=(pi*Da^2*va*e)/4; %deposition rate
vol=DR*t;%deposited material volume
anchow=C*2+0.003; %weld width
hw=vol/(lw*anchow);%weld height
filas=2;%layers of elements per weld layer

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%Ranges depending on factor C%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

if C>=0.005
    weldn=12;
end
if C<0.005 && C>=0.003
    weldn=10;
end
if C<0.003
    weldn=8;
end
sizey=hw/filas; %weld element size in y axis
sizex=anchow/weldn; %weld element size in x axis
sizew=sizex; %weld element size in w axis

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%WELD LAYERS GEOMETRY%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%vertex1
AW1=[1,0,0,0];

%vertex2
AW2=weldn;

AW=[AW1;
   AW2,anchow,0,0];

% vertex 3
AW3=weldn*filas+1;

AW=[AW;
    AW3,0,hw,0];
```

```
% vertex 4
AW4=weldn*filas+weldn;

AW=[AW;
   AW4,anchow,hw,0];
% vertex 5
AW5=round((lw/sizew))*AW4+1;
lw=round((lw/sizew))*sizew; %length recalculation according to element size

AW=[AW;
    AW5,0,0,lw];
%vertex6
AW6=AW5+weldn-1;

AW=[AW;
    AW6,anchow, 0,lw];
%vertex7
AW7=AW5+(weldn*filas);

AW=[AW;
    AW7, 0, hw,lw];
%vertex8
AW8=AW5+weldn*filas+weldn-1;

AW=[AW;
    AW8,anchow,hw,lw];

tiempo=lw/vp; %time for one layer
tiempo=roundn(tiempo,-2);
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%BASE GEOMETRY%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%
if bh>=0.01
    brows=6; %layers of element in the base
end
if bh<0.01 && bh>=0.007
    brows=5;
end
if bh<0.007
    brows=3;
end
bnode=round(bw/sizex); %number of nodes across the base

ely=bh/brows; %base element size in y axis
elx=bw/bnode; %base element size in X axis
elz=elx; %base element size in z axis
%vertex1
A1=[1,0,0,0];

%vertex2
A2=bnode;

A=[A1;
   A2,bw,0,0];
%vertex3
A3=bnode*brows+1;

A=[A;
```

```
    A3,0,bh,0];
%vertex4
A4=bnode*brows+bnode;

A=[A;
   A4,bw,bh,0];

%vertex5
A5=round((bl/elz))*A4+1;
bl=round((bl/elz))*elz;
A=[A;
    A5,0,0,bl];

%vertex6
A6=A5+bnode-1;

A=[A;
    A6,bw, 0,bl];
%vertex7
A7=A5+(bnode*brows);

A=[A;
    A7, 0, bh,bl];
%vertex8
A8=A5+bnode*brows+bnode-1  ;

A=[A;
    A8,bw,bh,bl];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%VERIFY IS THERMAL MODEL IS COMPLETED%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
st1 = Name;
st2 = '.sta';
st = strcat(sav,'\',st1,st2);
fid  = fopen(st,'r');
f=fread(fid,'*char')';
fclose(fid);
f = strrep(f,'U ','');
st = strcat(sav,'\','einc.txt');
fid  = fopen(st,'w');
fprintf(fid,'%s',f);
fclose(fid);
%%
st = strcat(sav,'\','einc.txt');
P=importdata(st);
inst=P.data;
[m,n]=size(inst);
I=[];
for j=1:m-1;
    if (inst(j,1)-inst(j+1,1)~=0)
        I1=inst(j,1:n);
        I=[I;
            I1];
    end
end
I=[I;
    inst(m,1:n)];
[m,n]=size(I);
einc=I(3 : 2 : m, 1:2);
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%EXCEL FILES DATA COLLECTION%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

s1 = BASEMATERIAL; %collect base material data
s2 = '.xlsx';
s3=sav;
s = strcat(s3,'\',s1,s2);
propiedades_base=xlsread(s);
[fila_prop,columna_prop]=size(propiedades_base);

a1=propiedades_base(:,1);
speceficiheat_temp= a1(~isnan(a1));
a2=propiedades_base(:,2);
speceficiheat_value=a2(~isnan(a2));
specificheat_base=[speceficiheat_value,speceficiheat_temp];

a3=propiedades_base(:,4);
density_temp= a3(~isnan(a3));
a4=propiedades_base(:,5);
density_value=a4(~isnan(a4));
density_base=[density_value,density_temp];
%
a5=propiedades_base(:,7);
conductivity_temp= a5(~isnan(a5));
a6=propiedades_base(:,8);
conductivity_value=a6(~isnan(a6));
conductivity_base=[conductivity_value,conductivity_temp];

a7=propiedades_base(:,10);
elasticity_value= a7(~isnan(a7));
a8=propiedades_base(:,11);
elasticity_poison= a8(~isnan(a8));
a9=propiedades_base(:,12);
elasticity_temp= a9(~isnan(a9));
elasticity_base=[elasticity_value,elasticity_poison,elasticity_temp];

a10=propiedades_base(:,14);
yieldstress_temp= a10(~isnan(a10));
a11=propiedades_base(:,15);
yieldstress_value= a11(~isnan(a11));
yieldstress_base=[yieldstress_value,yieldstress_temp];

a11=propiedades_base(:,17);
expansion_value= a11(~isnan(a11));
expansion_base=[expansion_value];

%%
s1 = WELDMATERIAL; %collect weld material data
s2 = '.xlsx';
s3=sav;
s = strcat(s3,'\',s1,s2);
propiedades_suelda=xlsread(s);
[fila_prop,columna_prop]=size(propiedades_suelda);

a1=propiedades_suelda(:,1);
speceficiheat_temp= a1(~isnan(a1));
a2=propiedades_suelda(:,2);
```

```
speceficiheat_value=a2(~isnan(a2));
specificheat_weld=[speceficiheat_value,speceficiheat_temp];

a3=propiedades_suelda(:,4);
density_temp= a3(~isnan(a3));
a4=propiedades_suelda(:,5);
density_value=a4(~isnan(a4));
density_weld=[density_value,density_temp];
%
a5=propiedades_suelda(:,7);
conductivity_temp= a5(~isnan(a5));
a6=propiedades_suelda(:,8);
conductivity_value=a6(~isnan(a6));
conductivity_weld=[conductivity_value,conductivity_temp];

a7=propiedades_suelda(:,10);
elasticity_value= a7(~isnan(a7));
a8=propiedades_suelda(:,11);
elasticity_poison= a8(~isnan(a8));
a9=propiedades_suelda(:,12);
elasticity_temp= a9(~isnan(a9));
elasticity_weld=[elasticity_value,elasticity_poison,elasticity_temp];

a10=propiedades_suelda(:,14);
yieldstress_temp= a10(~isnan(a10));
a11=propiedades_suelda(:,15);
yieldstress_value= a11(~isnan(a11));
yieldstress_weld=[yieldstress_value,yieldstress_temp];

a11=propiedades_suelda(:,17);
expansion_value= a11(~isnan(a11));
expansion_weld=[expansion_value];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%INPUT FILE CREATION FOR ABAQUS%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
s1 = Name;
s2 = '_mechanical.inp';
s3=sav;
s = strcat(s3,'\',s1,s2);

fi = fopen(s, 'wt');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%FILE HEADING%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

fprintf(fi,'%s\n%s\n%s\n%s\n','*HEADING','*PREPRINT, MODEL=NO,
HISTORY=NO,ECHO=NO','*FILE FORMAT, ZERO INCREMENT','*RESTART, WRITE,
FREQUENCY=5');
fprintf(fi, '%s%g%s%g%s%g%s\n','**Base(',bl*100,' x ',bh*100,' x
',bw*100,')[cm]');
fprintf(fi, '%s%g%s%g%s%g%s\n','**WELDING PARAMETERS(v=',vp*60,'[m/min],
Voltage=',voltaje,'[volts], I=',curr,'[Amp]');
fprintf(fi, '%s%g%s%g%s%g%s\n','**Wire Diameter=',Da*100,'[cm], Wire
speed=',va*60,'[m/min], Welding lenth=',lw*100,'[cm]');
fprintf(fi, '%s%g%s%s%g%s\n','**Heat input=',(10^-
3)*voltaje*curr/vp,'[kJ/m], ','Distribution factor C=',C*1000,'[mm]');
fprintf(fi, '%s\n','**---------------------------');
for k=0:PAS
    if k==0
```

```
        fprintf(fi, '%s\n%s\n','*Part, name=Base-Part','*End Part');
    else
        fprintf(fi, '%s%g%s\n%s\n','*Part, name=Weld',k,'-Part','*End 
Part');
        fprintf(fi, '%s\n','**');
    end
end
fprintf(fi, '%s\n','**---------------------------');
fprintf(fi, '%s\n','*Assembly, name=Assembly');
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','**---------------------------');

for j=0:PAS
    if j==0
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%BASE NODES/GEOMETRY%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        fprintf(fi, '%s\n','*Instance, name=Base-Part-1, part=Base-Part');
        fprintf(fi, '%s\n','*NODE, NSET=LIMITES');
        for k=1:length(A)
        fprintf(fi,'%g,%g,%g,%g\n', A(k,1),A(k,2),A(k,3),A(k,4));
        end
        for k=1:8
            fprintf(fi, '%s%g \n %g\n','*NSET,NSET=N',A(k,1),A(k,1));
        end
    fprintf(fi, '%s\n','**---------------------------');
    fprintf(fi, '%s\n
%s%g,%s%g,%g,%g\n','*NFILL,NSET=BOTO','N',A(1,1),'N',A(2,1),bnode-1,1);
    fprintf(fi, '%s\n
%s%g,%s%g,%g,%g\n','*NFILL,NSET=TOPO','N',A(3,1),'N',A(4,1),bnode-1,1);
    fprintf(fi, '%s\n
%s%g,%s%g,%g,%g\n','*NFILL,NSET=BOTF','N',A(5,1),'N',A(6,1),bnode-1,1);
    fprintf(fi, '%s\n
%s%g,%s%g,%g,%g\n','*NFILL,NSET=TOPF','N',A(7,1),'N',A(8,1),bnode-1,1);

    fprintf(fi, '%s\n','**---------------------------');
    fprintf(fi, '%s\n
%s,%s,%g,%g\n','*NFILL,NSET=BACK','BOTO','TOPO',brows,bnode);
    fprintf(fi, '%s\n
%s,%s,%g,%g\n','*NFILL,NSET=FRONT','BOTF','TOPF',brows,bnode);
    fprintf(fi, '%s\n
%s,%s,%g,%g\n','*NFILL,NSET=_Basemetal','BACK','FRONT',round((bl/elz)),bnod
e*(brows+1));

    fprintf(fi, '%s\n','**---------------------------');

    fprintf(fi, '%s\n
%s%g,%g,%g,%g,%g,%g\n','*ELEMENT,TYPE=C3D8,ELSET=MASTER','1,1,2,',bnode+2,b
node+1,A4+1,A4+2,A4+bnode+2,A4+bnode+1);
    fprintf(fi,
'%s\n%s%g,%s%g,%g,%g,%g,%g,%g\n','*ELGEN,ELSET=_Basemetal','1,',bnode-
1,'1,1,',brows,bnode,bnode-1,round((bl/elz)),A4,(bnode-1)*brows);

    fprintf(fi, '%s\n','**');
    fprintf(fi,
'%s\n%g,%g,%g,%g,%g,%g,%g,%g,%g,%g\n%s\n','*ELGEN,ELSET=_abajo',1,bnode-
1,1,1,1,bnode,bnode-1,round((bl/elz)),bnode*(brows+1),(bnode-
1)*brows,'*NSET,NSET=_abajo,ELSET=_abajo');
    fprintf(fi, '%s\n','**');
```

```
    fprintf(fi,
'%s\n%g,%g,%g,%g,%g,%g,%g,%g,%g,%g\n%s\n','*ELGEN,ELSET=_arriba',(bnode-
1)*(brows-1)+1,bnode-1,1,1,1,bnode,bnode-
1,round((bl/elz)),bnode*(brows+1),(bnode-
1)*brows,'*NSET,NSET=_arriba,ELSET=_arriba');
    fprintf(fi, '%s\n','**');

    fprintf(fi, '%s\n%s\n','*Solid Section, elset=_Basemetal,
material=BASEMATERIAL',',');
    fprintf(fi, '%s\n','*End Instance');

    fprintf(fi, '%s\n','**---------------------------');

    else
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%WELD NODES/GEOMETRY %%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

        fprintf(fi, '%s%g%s%g%s%g%s\n','*Instance, name=Weld',j,'-Part-
',j,', part=Weld',j,'-Part');
        fprintf(fi,'%g,%g,%g\n',bw/2-anchow/2,bh+(j-1)*hw,bl/2-lw/2);
        fprintf(fi, '%s\n','*NODE, NSET=LIMITES1');

        for k=1:length(AW)
            fprintf(fi,'%g,%g,%g,%g\n', AW(k,1),AW(k,2),AW(k,3),AW(k,4));
        end
        for k=1:8
            fprintf(fi, '%s%g \n %g\n','*NSET,NSET=N',AW(k,1),AW(k,1));
        end
        fprintf(fi, '%s\n','**---------------------------');
        fprintf(fi, '%s\n
%s%g,%s%g,%g,%g\n','*NFILL,NSET=BOTO','N',AW(1,1),'N',AW(2,1),weldn-1,1);
        fprintf(fi, '%s\n
%s%g,%s%g,%g,%g\n','*NFILL,NSET=TOPO','N',AW(3,1),'N',AW(4,1),weldn-1,1);
        fprintf(fi, '%s\n
%s%g,%s%g,%g,%g\n','*NFILL,NSET=BOTF','N',AW(5,1),'N',AW(6,1),weldn-1,1);
        fprintf(fi, '%s\n
%s%g,%s%g,%g,%g\n','*NFILL,NSET=TOPF','N',AW(7,1),'N',AW(8,1),weldn-1,1);


        fprintf(fi, '%s\n','**---------------------------');
        fprintf(fi, '%s\n
%s,%s,%g,%g\n','*NFILL,NSET=BACK','BOTO','TOPO',filas,weldn);
        fprintf(fi, '%s\n
%s,%s,%g,%g\n','*NFILL,NSET=FRONT','BOTF','TOPF',filas,weldn);
        fprintf(fi, '%s%g\n
%s,%s,%g,%g\n','*NFILL,NSET=_Weld',j,'BACK','FRONT',round((lw/sizew)),weldn
*(filas+1));

        fprintf(fi, '%s\n','**---------------------------');

        fprintf(fi, '%s\n
%s%g,%g,%g,%g,%g,%g\n','*ELEMENT,TYPE=C3D8,ELSET=MASTER','1,1,2,',weldn+2,w
eldn+1,AW4+1,AW4+2,AW4+weldn+2,AW4+weldn+1);
        fprintf(fi,
'%s%g\n%s%g,%s%g,%g,%g,%g,%g,%g\n','*ELGEN,ELSET=_Weld',j,'1,',weldn-
1,'1,1,',filas,weldn,weldn-1,round((lw/sizew)),AW4,(weldn-1)*filas);

        fprintf(fi, '%s\n','**');
```

```
        fprintf(fi,
'%s%g\n%g,%g,%g,%g,%g,%g,%g,%g,%g,%g\n%s%g%s%g\n','*ELGEN,ELSET=_LowerWeld'
,j,1,weldn-1,1,1,1,weldn,weldn-1,round((lw/sizew)),weldn*(filas+1),(weldn-
1)*filas,'*NSET,NSET=_LowerWeld',j,',ELSET=_LowerWeld',j);

        fprintf(fi, '%s\n','**');
        fprintf(fi,
'%s%g\n%g,%g,%g,%g,%g,%g,%g,%g,%g,%g\n%s%g%s%g\n','*ELGEN,ELSET=_UpperWeld'
,j,(weldn-1)*(filas-1)+1,weldn-1,1,1,1,weldn,weldn-
1,round((lw/sizew)),weldn*(filas+1),(weldn-
1)*filas,'*NSET,NSET=_UpperWeld',j,',ELSET=_UpperWeld',j);

        fprintf(fi, '%s\n','**');

        fprintf(fi, '%s%g%s\n%s\n','*Solid Section, elset=_Weld',j,',
material=WELDMATERIAL',',');
        fprintf(fi, '%s\n','*End Instance');
    end
    end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%SURFACES CREATION%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for k=0:PAS
    if k==0
        fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','**++++++++++++++++++++++++++++++++++++++++++++++');
fprintf(fi, '%s\n','**');
fprintf(fi,
'%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n','*Nset,nset=Base
metal,instance=Base-Part-
1','_Basemetal','*Elset,elset=Basemetal,instance=Base-Part-
1','_Basemetal','*Elset,elset=arriba,instance=Base-Part-
1','_arriba','*SURFACE,type=ELEMENT,NAME=arriba','arriba,S5','
*Nset,nset=abajo,instance=Base-Part-
1','_abajo','*Elset,elset=abajo,instance=Base-Part-
1','_abajo','*SURFACE,type=ELEMENT,NAME=abajo','abajo,S3');
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n%g,%g,%g\n%s\n%s\n%s\n','*ELSET,ELSET=atras,instance=Base-
Part-1,GENERATE',1,(bnode-
1)*brows,1,'*SURFACE,type=ELEMENT,NAME=atras','atras,S1','*NSET,NSET=atras,
ELSET=atras');
fprintf(fi, '%s\n','**');
fprintf(fi,
'%s\n%g,%g,%g\n%s\n%s\n%s\n','*ELSET,ELSET=adelante,instance=Base-Part-
1,GENERATE',round((bl/elz))*((bnode-1)*brows)-(((bnode-1)*brows)-
1),round((bl/elz))*((bnode-
1)*brows),1,'*SURFACE,type=ELEMENT,NAME=adelante','adelante,S2','*NSET,NSET
=adelante,ELSET=adelante');
fprintf(fi, '%s\n','**');
fprintf(fi,
'%s\n%g,%g,%g\n%s\n%s\n%s\n','*ELSET,ELSET=izquierda,instance=Base-Part-
1,GENERATE',1,round((bl/elz))*((bnode-1)*brows)-(bnode-2),bnode-
1,'*SURFACE,type=ELEMENT,NAME=izquierda','izquierda,S6','*NSET,NSET=izquier
da,ELSET=izquierda');
fprintf(fi, '%s\n','**');
fprintf(fi,
'%s\n%g,%g,%g\n%s\n%s\n%s\n','*ELSET,ELSET=derecha,instance=Base-Part-
1,GENERATE',bnode-1,round((bl/elz))*((bnode-1)*brows),bnode-
```

```
1,'*SURFACE,type=ELEMENT,NAME=derecha','derecha,S4','*NSET,NSET=derecha,ELS
ET=derecha');
    else
        fprintf(fi, '%s\n','**');
        fprintf(fi,
'%s\n','**++++++++++++++++++++++++++++++++++++++++++++++');
        fprintf(fi, '%s\n','**');
        fprintf(fi,
'%s%g%s%g%s%g\n%s%g\n%s%g%s%g%s%g\n%s%g\n','*Nset,nset=Weld',k,',instance=W
eld',k,'-Part-',k,'_Weld',k,'*Elset,elset=Weld',k,',instance=Weld',k,'-
Part-',k,'_Weld',k);
        fprintf(fi, '%s\n','**');
        fprintf(fi,
'%s%g%s%g%s%g\n%s%g\n%s%g%s%g%s%g\n%s%g\n%s%g%s\n%s%g%s\n','*Nset,nset=Lowe
rWeld',k,',instance=Weld',k,'-Part-
',k,'_LowerWeld',k,'*Elset,elset=LowerWeld',k,',instance=Weld',k,'-Part-
',k,'_LowerWeld',k,'*Surface, type=ELEMENT, name=LowerWeld',k,'-
Surface','LowerWeld',k,',S3');
        fprintf(fi, '%s\n','**');
        fprintf(fi,
'%s%g%s%g%s%g\n%s%g\n%s%g%s%g%s%g\n%s%g\n%s%g%s\n%s%g%s\n','*Nset,nset=Uppe
rWeld',k,',instance=Weld',k,'-Part-
',k,'_UpperWeld',k,'*Elset,elset=UpperWeld',k,',instance=Weld',k,'-Part-
',k,'_UpperWeld',k,'*Surface, type=ELEMENT, name=UpperWeld',k,'-
Surface','UpperWeld',k',', S5');
        fprintf(fi, '%s\n','**');
        fprintf(fi, '%s\n','**');
        fprintf(fi,
'%s%g%s%g%s%g%s\n%g,%g,%g\n%s%g\n%s%g%s\n%s%g%s%g\n','*ELSET,ELSET=atras',k
,',instance=Weld',k,'-Part-',k,',GENERATE',1,(weldn-
1)*filas,1,'*SURFACE,type=ELEMENT,NAME=atras',k,'atras',k,',S1','*NSET,NSET
=atras',k,',ELSET=atras',k);
        fprintf(fi, '%s\n','**');
        fprintf(fi,
'%s%g%s%g%s%g%s\n%g,%g,%g\n%s%g\n%s%g%s\n%s%g%s%g\n','*ELSET,ELSET=adelante
',k,',instance=Weld',k,'-Part-',k,',GENERATE',round((lw/sizew))*((weldn-
1)*filas)-(((weldn-1)*filas)-1),round((lw/sizew))*((weldn-
1)*filas),1,'*SURFACE,type=ELEMENT,NAME=adelante',k,'adelante',k,',S2','*NS
ET,NSET=adelante',k,',ELSET=adelante',k);
        fprintf(fi, '%s\n','**');
        fprintf(fi,
'%s%g%s%g%s%g%s\n%g,%g,%g\n%s%g\n%s%g%s\n%s%g%s%g\n','*ELSET,ELSET=izquierd
a',k,',instance=Weld',k,'-Part-',k,',GENERATE',1,round((lw/sizew))*((weldn-
1)*filas)-(weldn-2),weldn-
1,'*SURFACE,type=ELEMENT,NAME=izquierda',k,'izquierda',k,',S6','*NSET,NSET=
izquierda',k,',ELSET=izquierda',k);
        fprintf(fi, '%s\n','**');
        fprintf(fi,
'%s%g%s%g%s%g%s\n%g,%g,%g\n%s%g\n%s%g%s\n%s%g%s%g\n','*ELSET,ELSET=derecha'
,k,',instance=Weld',k,'-Part-',k,',GENERATE',weldn-
1,round((lw/sizew))*((weldn-1)*filas),weldn-
1,'*SURFACE,type=ELEMENT,NAME=derecha',k,'derecha',k,',S4','*NSET,NSET=dere
cha',k,',ELSET=derecha',k);
    end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%MODEL CONSTRAINTS%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fprintf(fi, '%s\n','**');
```

```
fprintf(fi, '%s\n%s\n%s\n','** Constraint: Tie1-Constraint','*Tie,
name=Tie1-Constraint, adjust=yes','LowerWeld1-Surface, arriba');
fprintf(fi, '%s\n','**');
for k=2:PAS
fprintf(fi, '%s%g%s\n%s%g%s%g%s\n','*Tie, name=Tie',k,'-Constraint,
adjust=yes','LowerWeld',k,'-Surface,UpperWeld',k-1,'-Surface');
end
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','*End Assembly');


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%MATERIAL PROPERTIES%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fprintf(fi, '%s\n','**Materials');
fprintf(fi, '%s\n','**---------------------------');
fprintf(fi, '%s\n','*MATERIAL,NAME=BASEMATERIAL');

fprintf(fi, '%s\n','*ELASTIC');
[filaelasticity,columnaelasticity]=size(elasticity_base);
for k=1:filaelasticity
    if columnaelasticity==2
    fprintf(fi, '%g,%g\n',elasticity_base(k,1),elasticity_base(k,2));
    else
        fprintf(fi,
'%g,%g,%g\n',elasticity_base(k,1),elasticity_base(k,2),elasticity_base(k,3)
);
    end

end

fprintf(fi, '%s\n','*PLASTIC');
[filayieldstress,columnayieldstress]=size(yieldstress_base);
for k=1:filayieldstress
    fprintf(fi,
'%g,%g,%g\n',yieldstress_base(k,1),0,yieldstress_base(k,2));

end


fprintf(fi, '%s\n','*EXPANSION');
[filaexpansion,columnaexpansion]=size(expansion_base);
for k=1:filaexpansion
    fprintf(fi, '%g\n',expansion_base(k,1));
end


fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','*MATERIAL,NAME=WELDMATERIAL');

fprintf(fi, '%s\n','*ELASTIC');
[filaelasticity,columnaelasticity]=size(elasticity_weld);
for k=1:filaelasticity
    if columnaelasticity==2
    fprintf(fi, '%g,%g\n',elasticity_weld(k,1),elasticity_weld(k,2));
    else
        fprintf(fi,
'%g,%g,%g\n',elasticity_weld(k,1),elasticity_weld(k,2),elasticity_weld(k,3)
);
```

```
    end

end

fprintf(fi, '%s\n','*PLASTIC');
[filayieldstress,columnayieldstress]=size(yieldstress_weld);
for k=1:filayieldstress
    fprintf(fi,
'%g,%g,%g\n',yieldstress_weld(k,1),0,yieldstress_weld(k,2));

end


fprintf(fi, '%s\n','*EXPANSION');
[filaexpansion,columnaexpansion]=size(expansion_weld);
for k=1:filaexpansion
    fprintf(fi, '%g\n',expansion_weld(k,1));
end
fprintf(fi, '%s\n','**');





fprintf(fi, '%s\n','** BOUNDARY CONDITIONS');
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','**Name: Disp-BC-1 Type:
Symmetry/Antisymmetry/Encastre');
fprintf(fi, '%s\n','*Boundary');
fprintf(fi, '%s\n','ABAJO, ENCASTRE');





%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%INITIAL CONDITIONS%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n%s%g\n','*INITIAL
CONDITIONS,TYPE=TEMPERATURE','Basemetal,',Tinitial);

for k=1:PAS
    fprintf(fi, '%s%g,%g\n','Weld',k,Tinitial);
end
fprintf(fi, '%s\n','*PHYSICAL CONSTANTS,ABSOLUTE ZERO=-273.16,STEFAN
BOLTZMANN=5.66E-8');
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','**----------------------------------------------');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%STEPS/SIMULATION CONDITIONS/SUBROUTINES%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fprintf(fi, '%s\n','*Step, name=Remove-Step, nlgeom=NO');
fprintf(fi, '%s\n%s\n','*Static','0.,0.001,1.E-6,,1.E-4');
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','*Model Change, remove');
for k=1:PAS
    fprintf(fi, '%s%g,','Weld',k);
```

```
end
fprintf(fi, '\n%s\n','**');
fprintf(fi, '%s\n','**OUTPUT REQUESTS');
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','*Restart, write, frequency=0');
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','**FIELD OUTPUT: F-Output-1');
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n%s\n%s\n','*Output, field, variable=PRESELECT','*Output,
history, frequency=0','*End Step');

fprintf(fi, '%s\n','**-----------------------------------------------');



for j=1:PAS
fprintf(fi, '%s%g%s\n%s\n%s\n%s\n%s\n','*Step, name=Add',j,'-Step,
nlgeom=NO','*Static','0.,0.001,1.E-6,,1.E-4','**','*Model Change, add');
fprintf(fi, '%s%g,','Weld',j);
fprintf(fi, '\n%s\n','**');
fprintf(fi, '%s\n','**OUTPUT REQUESTS');
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','*Restart, write, frequency=0');
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','** FIELD OUTPUT: F-Output-1');
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','*Output, field, variable=PRESELECT');
fprintf(fi, '%s\n','*Output, history, frequency=0','*End Step');
fprintf(fi, '%s\n','** ---------------------------------------------------
------------');

fprintf(fi, '%s%g\n','** STEP: Step-',j);
fprintf(fi, '%s%g%s\n','*Step, name=Step-',j,', nlgeom=NO, INC=4000');
fprintf(fi, '%s\n','**');
if j==PAS
    fprintf(fi, '%s\n%s%g,%s\n','*Static','0.001,',tiempo+tiempoextra,'1.E-
6,,1.E-4');
else
fprintf(fi, '%s\n%s%g,%s\n','*Static','0.001,',tiempo,'1.E-6,,1.E-4');
end
fprintf(fi, '%s%s%s%g%s%g%s%g\n','*Temperature, file=',Name,',
bstep=',einc(j,1),', binc=1, estep=',einc(j,1),', einc=',einc(j,2));

fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','**OUTPUT REQUESTS');
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','*Restart, write, frequency=0');
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','** FIELD OUTPUT: F-Output-1');
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','*Output, field, variable=PRESELECT');
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','** HISTORY OUTPUT: H-Output-1');
fprintf(fi, '%s\n','**');
fprintf(fi, '%s\n','*Output, history, variable=PRESELECT');
fprintf(fi, '%s\n','*End Step');
fprintf(fi, '%s\n','** ---------------------------------------------------
------------');
end

fclose(fi);
```

APPENDIX C: MATLAB LINKING ABAQUS CODE

```
if check==1
      %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
      %%%%%%%COMMANDS FILE .BAT%%%%%%%%
      %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

        dire1='Mat2Aba.bat';
        dire = strcat(sav,'\',dire1);
        fi = fopen(dire, 'wt');
      %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
      %%%%%%SETS THE DIRECTORY FOLDER OF INPUT FILES%%%%%%%%
      %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

        fprintf(fi, '%s %s\n%s%s %s%s%s','cd',sav,'abaqus
        job=',m,'user=',m,'.for');

      %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
      %%%%%%COMMANDS TO USE FORTRAN COMPILER TO USE SUBROUTINES%%%%%%%%
      %%%%%%%%%%%%%%%%%%%%%%%VISUAL STUDIO 14%%%%%%%%%%%%%%%%%%%%%%%%%
        str1 = '"C:\Program Files (x86)\Microsoft Visual Studio
        14.0\VC\bin\amd64\vcvars64.bat"';

      %%%%%%%%%%%%%%%%%%%%%%%INTEL FORTRAN%%%%%%%%%%%%%%%%%%%%%%%%%%
        str2 = '"C:\Program Files
        (x86)\IntelSWTools\compilers_and_libraries_2016.1.146\windows\bin\i
        fortvars.bat"';
        str3 = 'intel64 vs2015';
        str4 = 'Mat2Aba.bat';
        str5= strcat(sav,'\',str4);

      %%%%%CONCATENATE STRINGS %%%%%%%%
       comando = sprintf('%s & %s %s & %s', str1, str2, str3, str5);

      %%%%%WRITE STRING COMANDO IN COMMAND PROMPT %%%%%%%%
        dos(comando);
            set(handles.OutputData,'string','Abaqus Thermal Model is
      running');
```