An Implicit Algorithm Based on Continuous Moving Least Square to Simulate Material Mixing in Friction Stir Welding Process

Abdelaziz Timesli,1,2 Bouazza Braikat,1 Hassane Lahmam,1 and Hamid Zahrouni2

1 Laboratoire d’Ingénierie et Matériaux LIMAT, Faculté des Sciences Ben M’Skik, Université Hassan II Mohammedia-Casablanca, Sidi Othman, BP 7955, Casablanca, Morocco
2 Laboratoire d’Etude des Microstructures et de Mécanique des Matériaux LEM3, CNRS UMR 7239, Université de Lorraine, Île du Saulcy, 57045 Metz Cedex 01, France

Correspondence should be addressed to Abdelaziz Timesli; abdelaziz.timesli@univ-lorraine.fr

Received 25 June 2013; Revised 16 September 2013; Accepted 30 September 2013

1. Introduction

Friction stir welding (FSW) [1] is a solid state welding process used to join two pieces (plates, tubes, spheres, etc.) of aluminum alloys. Joining two workpieces by FSW consists in heat generation mainly due to the shoulder and material mixing thanks to the pin that provokes both extremely high plastic deformation and also a high heat generation.

Numerical simulation of friction stir welding process is important in industry to control the product quality and optimize the fabrication cost and avoid the traditional tests. It has been investigated by several authors considering thermal or thermomechanical framework [2–8]. The difficulty in modeling this welding type resides to large deformations generated by the material mixing. Different formulations have been proposed in Eulerian, Lagrangian, or Arbitrary Lagrangian Eulerian frameworks. Eulerian formulation is appropriate to describe material flow for large deformations thanks to fixed mesh grid but this formulation cannot be used for problems involving free surfaces. Lagrangian formulation is well adapted for history dependence in solids and for simulation of material flow with free surfaces but mesh distortions require special procedure as remeshing. The ALE formulation has been proposed to benefit from both advantages of Eulerian and Lagrangian formulations [3]. In this technique, one has to describe the motion of the mesh and material particles separately with respect to a reference domain. ALE has been applied successfully in many fields and particularly in metal forming processes. Despite the intensive development of these techniques, material mixing in FSW process remains very difficult to achieve numerically and especially when using finite element method [2, 3].

An alternative solution for the simulation of this process is the use of meshless methods. Indeed, these techniques are developed for several decades to solve partial differential equations. They aim to avoid difficulties related to the mesh discretization and they have been intensively developed since the 90s. To our knowledge, smoothed particle hydrodynamics (SPH) was the first meshless method presented in 1977 by Lucia and Gingold and Monaghan for astrophysical problems.
Another class of meshless methods called diffuse element method was proposed by Nayroles et al. [11, 12]. This method is based on local and moving least square approximation and a Galerkin discretization scheme. Belytschko et al. developed in 1994 element-free Galerkin method (EFG) [13].

The SPH method based on the notion of particles is best suited for large deformation problems [14] and the MLS approach [15] provides a better consistency for SPH. The first reason is that MLS approach uses monomial basis functions to ensure that a local polynomial expansion best fits the surrounding data points. The second reason is that there is less oscillation of the interpolating polynomial between particles with the MLS which is the source of the tensile instability [16]. In addition, the boundary integrals were dropped in the derivation of the particle equations for SPH, so corrections are needed near boundaries [17].

The aim of the proposed work is to implement an implicit algorithm to simulate material mixing in friction stir welding. This algorithm is based on the continuous moving least square method (CMLS) [18] which combines the advantages of both approaches SPH and MLS. CMLS is formulated here in the Lagrangian framework, using the strong form of partial differential equations. This framework allows one to avoid the drawback related to numerical integration. Thermal and mechanical fields are directly attached to the points used for the domain discretization by the MLS shape functions [19]. In this work, the boundary conditions are imposed with the boundary point collocation method [20], where collocation points are coincident with nodes along the boundary where the solutions are imposed. This allows one to easily simulate large deformations and material mixing around the welding tool.

The layout of this paper is as follows. In Section 2, we present the strong form of the governing equations. In Section 3, we detail the main idea of the CMLS method and we present space and time discretization procedures. In Section 4, we present the solution strategy adopted in this work. In Section 5, numerical application is proposed and a comparison with the industrial code Fluent is given to validate our model. This code is based on an Eulerian formulation and a finite volume discretization. Finally, some conclusions are presented in Section 6.

2. Governing Equations

Friction stir welding is a complex thermomechanical process that requires accurate knowledge of the relations existing between the main parameters of the process such as plunge depth, travel speed, rotation speed, thermomechanical properties, and tool geometry. In many contributions, the mixing zone is considered as a high viscosity incompressible fluid and the flow is obtained using computational fluid dynamics. In some other cases, the deformation is modeled using solid mechanics and numerical methods to solve the resulting nonlinear problem to compute the different variables as velocity, temperature, and so forth. In the present work, we consider a mechanical problem with a constitutive law that depends only on the equivalent strain rate. The resulting problem is then described by the conservation laws including the following.

Mass conservation:

$$\text{div} \ (V) = 0.$$  \hspace{1cm} (1)

Motion equation:

$$\rho \frac{dV}{dt} = \text{div} \ (\sigma),$$ \hspace{1cm} (2)

where $\rho$ is the material density, $V$ is the velocity vector, $\sigma$ is the stress tensor, and the gravity is neglected. The stress tensor $\sigma$, under this assumption, is given by

$$\sigma = -pI + S,$$ \hspace{1cm} (3)

where $p$ is the hydrostatic pressure, $S$ is the deviatoric stress tensor, and $I$ is the unity tensor. To satisfy the condition of incompressibility numerically in (1), we introduce, in general, a pressure term penalized by a large factor noted $\lambda$. This means that the incompressibility condition (1) is replaced by a viscous law [21]:

$$p = -\lambda Tr (\dot{\varepsilon}),$$ \hspace{1cm} (4)

where $\dot{\varepsilon}$ is the strain rate tensor defined as

$$\dot{\varepsilon} = \frac{1}{2} \left( \nabla V + \nabla V^T \right).$$ \hspace{1cm} (5)

FSW is a process that generates large strains in which elastic strains are negligible. The advantage of this hypothesis is that the material can then be modeled as viscoplastic material:

$$S = 2\mu \dot{\varepsilon},$$ \hspace{1cm} (6)

where $\mu$ is the material viscosity. Taking into account (3), (4), and (6), the stress tensor is given by

$$\sigma = \lambda Tr (\dot{\varepsilon}) I + 2\mu \dot{\varepsilon}.$$ \hspace{1cm} (7)

For the viscosity, we choose a power law given by [22]

$$\mu = \frac{1}{3} kT^{n-1} \epsilon^{n-1},$$ \hspace{1cm} (8)

where $T$ is the temperature, $k$, $A$, and $n$ are the material properties, and $\bar{\varepsilon}$ is the equivalent strain rate given by

$$\bar{\varepsilon} = \sqrt{\frac{2}{3} \dot{\varepsilon} : \dot{\varepsilon}}.$$ \hspace{1cm} (9)

In this work, we focus our efforts on the mixing aspect that is difficult to achieve by the conventional methods such as finite element one. So, we assume that the temperature is constant and then the viscosity depends only on the equivalent strain rate ($\mu(T, \bar{\varepsilon}) = \mu(\bar{\varepsilon})$). Finally, the governing equations can be summarized as follows:

$$\rho \frac{dV}{dt} = \text{div} \ (\sigma),$$

$$\sigma = \lambda Tr (\dot{\varepsilon}) I + 2\mu \dot{\varepsilon},$$

$$\dot{\varepsilon} = \frac{1}{2} \left( \nabla V + \nabla V^T \right).$$
Initialization data: \{V_0^m\}, \{ρ^0\}, ...

\begin{align*}
\text{Loop over time} & \quad n = n + 1 \\
\text{Prediction} & \\
\text{Consider } \{V^n\} \text{ as the problem solution} & \\
\text{Problem solving } (\{M\} - Δt[K(\{V^n\})])\{V^{n+1}_0\} = [M]\{V^n\} & \\
\text{Correction} & \\
\text{Problem solving } (\{M\} - Δt[K(\{V^{n+1}_k\})])\{V^{n+1}_k\} = [M]\{V^n\} & \\
\text{Test } \frac{||V^{n+1}_k - V^{n+1}_{k-1}||}{||V^{n+1}_k||} \leq δ & \\
\text{Shape functions update} & \\
\text{Neighboring points update} & \\
\text{Coordinate update} & \{ρ^{n+1}\} = \{ρ^n\} + Δ\{V^{n+1}\} & \\
\text{Velocity vector update} & \{V^{n+1}\} = \{V^{n+1}_k\} & \\
\text{No} & \\
\text{Yes} & k = k + 1 &
\end{align*}

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{algorithm_sketch.png}
\caption{Global sketch of the proposed implicit algorithm.}
\end{figure}

\begin{align*}
\varepsilon &= \sqrt{\frac{2}{3} \dot{\varepsilon} : \dot{\varepsilon}}, \\
\mu &= \frac{1}{3} KT^{-4} \varepsilon^{-1}.
\end{align*} 

Problem (10) will be completed by boundary and initial conditions.

\section{3. Numerical Model}

\subsection{3.1. Principal of CMLS Technique}

Moving least square (MLS) technique belongs to the family of meshless methods. Since the pioneering work about diffuse elements presented by Nayroles et al. in 1992 [12], several contributions using this technique have been developed in many mathematical and mechanical fields [23, 24]. Continuous moving least square method (CMLS) is a modified version of MLS as shown in...
In this method, the approximation \( u^h \) of the function \( u \) is defined as a polynomial of order \( m \) but with nonconstant coefficients. The local approximation at the coordinate \( x \in \Omega \) is given by

\[
u^h(x) = \sum_{i=1}^{m} p_i(x) a_i(x) = \langle p(x) \rangle a(x), \tag{11}\]

where \( p(x) \) is the basis function of the spatial coordinates and \( m \) is the number of the basis functions. The vector coefficients \( \{a(x)\} \) are obtained by performing a weighted least square fit for the local approximation. This yields the continuous quadratic form

\[
J(x) = \frac{1}{2} \int_{\Omega} w(x-y, h) \left[ \langle p(y) \rangle a(x) - u(y) \right]^2 dy, \tag{12}\]

where \( \Omega \) represents the support domain of point \( x \), \( u(y) \) is the nodal parameter of \( u \) at \( x = y \), and \( w(x-y, h) \) is a weighting function with a compact support \( h \). The weight function plays an important role in the performance of the MLS approximation. The exponential function and spline functions are often used in practice; for example, we present a weighting function in the form of cubic spline [25]:

\[
w(r) = \begin{cases} 
\frac{10}{7\pi h^2} \left( 1 - \frac{3}{2}r^2 + \frac{3}{4}r^3 \right) & \text{for } r \leq 1 \\
\frac{10}{28\pi h^2} (2-r)^3 & \text{for } 1 < r \leq 2 \\
0 & \text{for } r > 2,
\end{cases} \tag{13}\]

where \( r = (x-y)/h \). The vector coefficients \( \{a(x)\} \) are determined in each point by minimizing the functional \( J(x) \). This minimization leads to the following set of linear equations for \( \{a(x)\} \):

\[
[A(x)] \{a(x)\} = \int_{\Omega} w(x-y, h) \{p(y)\} \{u(y)\} d\Omega, \tag{14}\]

where \( \{u\} \) is the vector that collects the nodal parameters of field function for all the points in the support domain. \( [A(x)] \) is called the weighted moment matrix given by

\[
[A(x)] = \int_{\Omega} w(x-y, h) \{p(y)\} \{p(y)\} d\Omega. \tag{15}\]
Solving \{a(x)\} from (14) and substituting it into (11), the CMLS approximants can be defined as

\[ u_h^h(x) = \sum_{i=1}^{N} T \{ p(x) \} [A(x)]^{-1} \{ p(y) \} w(x - y, h) |u(y)| d\Omega. \] (16)

The approximation defined by (16) is difficult to use due to its integral form. An equivalent discrete form of this equation constructed by a particle approximation is proposed. In this method, the approximation of (16) at an interpolated point of coordinate \( x \), in discrete notation, leads to the following approximation:

\[ u_h^h(x) = \sum_{i=1}^{N} \phi_i(x) u_i \Delta V_i, \] (17)

where \( N \) is the number of points in the support domain, \( u_i \) is the nodal parameter of \( u \) at \( x = x_i \) and \( \Delta V_i \) is the volume associated with point \( i \).

The number of points \( N \), chosen in the support domain, should be sufficient to ensure that the matrix \([A(x)]\) in (15) is invertible, so as to provide the interpolation stability. The choice of \( N \) depends on the nodal distribution and the number of basis functions. In order to ensure the existence of \([A(x)]^{-1}\) and a well-conditioned \([A(x)]\), we usually let \( N \gg m \) [26, 27]. Unfortunately, there is no theoretical best value of \( N \), and it has to be determined by numerical experiments.

Recalling the form of the approximation defined in (11), so

\[ u_h^h(x) = \sum_{i=1}^{N} \phi_i(x) u_i \Delta V_i, \] (18)
Finally, the CMLS shape functions $\phi(x)$ are defined as
\[ \phi_i(x) = \frac{T\{p(x)\}[A(x)]^{-1}\{p(x_i)\} w(x-x_i,h_i)}{\sum_{i=1}^{N} w(x-x_i,h_i)}. \] (19)

The CMLS shape function $\phi_i(x)$ will be continuous in the entire global domain, as long as the weight functions $w(r)$ are chosen properly. Field points included in this support domain are used to perform the CMLS approximation for the unknown function at this point.

In the case where the interpolated point is a particle, the SPH method is used for the calculation of the particle volume [18, 28] as follows:
\[ \Delta V(x_k) = \frac{1}{\sum_{i=1}^{N} w(x_i-x_k,h_k)}. \] (20)
where $k$ is the reference particle.

3.2. Algorithm for the Nearest Neighbors Search. In meshless methods, search for neighboring points in the influence domain is the most expensive procedure in terms of time consumption. The search of neighboring points must be performed for each point in the studied domain and many times during the computation. The naive technique consists in performing a double loop over all collocation points leading to a very expensive algorithm. Several methods are proposed in the literature to improve the performance of these algorithms [17, 29]. In this work, we have implemented an algorithm well adapted to the studied problem. Indeed, we have proposed to construct a fixed grid containing the domain subdivided into $n_r$ subdomains in the radial direction and $n_\theta$ subdomains along the circumferential direction. The size of each subdomain is noted $(dr,d\theta)$, where $dr$ is the...
Figure 6: Distribution of $\dot{\varepsilon}_{xy}$ component of the strain rate given by CMLS algorithm at different interpoints distances $dr = 5 \times 10^{-4}$ m, $dr = 3 \times 10^{-4}$ m, $dr = 2.5 \times 10^{-4}$ m, and $dr = 2.25 \times 10^{-4}$ m, respectively.

Table 1: Influence of the parameter $h$ on the results of the proposed algorithm.

<table>
<thead>
<tr>
<th>Support size</th>
<th>$h = 1.5dr$</th>
<th>$h = 2dr$</th>
<th>$h = 2.5dr$</th>
<th>$h = 3dr$</th>
<th>$h = 4dr$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum relative difference</td>
<td>40%</td>
<td>10%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
</tr>
</tbody>
</table>

Table 2: Influence of the weighting function on the results obtained by the proposed algorithm.

<table>
<thead>
<tr>
<th>Weighting function</th>
<th>Quadratic spline</th>
<th>Cubic spline</th>
<th>Quartic spline</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum relative difference</td>
<td>10%</td>
<td>5%</td>
<td>5%</td>
</tr>
</tbody>
</table>

The choice of $dr$ and $d\theta$ must depend on mesh density of collocation points in the domain and it can be considered as a user parameter. Note that the grid is fixed and each subdomain knows its neighboring subdomains. The procedure defining the neighbors of a given subdomain is executed only one time in the algorithm. For each subdomain, we have then the number of the subdomain and the list of its neighbors. The particular cases of the subdomains located at the domain edge are considered with a specific treatment. Each collocation point $P$ is defined by a given number of the point, its coordinates $(x,y)$, and the number of
the subdomain to which the considered point belongs (see Figure 1(b)). By using a simple mathematical operation, we can identify the number of the subdomain to which the considered point belongs. This operation consists in dividing the current position of the point by the subdomain size in radial or circumferential direction. Retaining only the integer part of the division, we obtain the position in the radial and circumferential directions of the subdomain to which the point belongs. All the points of this subdomain and of the neighboring subdomains are neighbors the considered point. The proposed algorithm does not require large computation time. At each end step, coordinates of collocation points and the corresponding subdomain are updated.

3.3. Space Discretization. The use of meshless methods to approximate the solution of the strong form of the mechanical problem leads to the following advantages: (i) the discretized equations can be obtained directly from the strong forms
of governing equations of the problem; (ii) no numerical integration is required; (iii) no mesh is used.

To apply this technique to the problem (10), the shape functions presented in Section 3.1 are used to approximate the velocity vector \( \{\mathbf{V}\} = T \langle \mathbf{u}, \nu \rangle \) in the bidimensional case, at any point using a set of points in a local support domain of the considered point:

\[
\{\mathbf{V}\} = \sum_{i=1}^{N} \begin{bmatrix} \phi_i \end{bmatrix} \begin{bmatrix} V_i \end{bmatrix} \Delta V_i, \tag{21}
\]

where \( \begin{bmatrix} \phi_i \end{bmatrix} \) is the matrix of shape functions of point \( i \) and \( \begin{bmatrix} V_i \end{bmatrix} \) is the velocity vector of point \( i \). Using (5) and (21), the strain rate vector can be obtained using the approximated velocity vector:

\[
\{\dot{\mathbf{e}}\} = \sum_{i=1}^{N} \begin{bmatrix} B_i \end{bmatrix} \begin{bmatrix} V_i \end{bmatrix} \Delta V_i, \tag{22}
\]

where \( \begin{bmatrix} B_i \end{bmatrix} \) is the matrix of derivatives of shape functions of point \( i \). Substituting (21) and (22) in (10) and assembling, we obtain nonlinear system defined by

\[
\begin{bmatrix} M \end{bmatrix} \begin{bmatrix} \dot{V} \end{bmatrix} - \begin{bmatrix} K(\{V\}) \end{bmatrix} \{V\} = \{0\}, \tag{23}
\]

where \( \begin{bmatrix} M \end{bmatrix} \) is the global mass matrix and \( \begin{bmatrix} K(\{V\}) \end{bmatrix} \) is a global matrix dependent of \( \{V\} \) which is the vector that collects the point velocities.
3.4. Time Integration. Using an Euler implicit scheme, the time discretization of the previous problem leads to the following nonlinear system in terms of the new unknown \( \{V_{n+1}\} \):

\[
\left([M] - \Delta t \left[K \left(\{V_{n+1}\}\right)\right]\right) \{V_{n+1}\} = [M] \{V_n\},
\]

(24)

where \( V_j \) represents the value of the unknown at time \( t_j = j \Delta t \).

4. Solution Strategy

Within the so-called iterative method [30], generally, a space discretization of this problem is carried out and the obtained discretized equations are solved iteratively. The nodal unknown is denoted by \( \{V_{n+1}\} \). In the first iteration, one assumes in the term \( K(\{V_{n+1}\}) \) the value of \( \{V_{n+1}\} = \{V_n\} \) to start with. Therefore, the right hand side becomes known. Next one iterates until the convergence criterion is satisfied (see Figure 2). This process is controlled by a tolerance \( \delta \). The iterative process will continue until

\[
\left\|\{V_{k+1}\} - \{V_{k}\}\right\| \leq \delta.
\]

(25)

The solution of the problem (24) is the limit of the sequence \( \{V_1^{n+1}, V_2^{n+1}, \ldots, V_k^{n+1}, \ldots\} \) which is the solution of the following recurrent algebraic systems:

\[
([M] - \Delta t \left[K \left(\{V_{k-1}\}\right)\right]) \{V_k^{n+1}\} = [M] \{V_k^n\}, \quad k \geq 1.
\]

(26)

5. Numerical Analysis and Applications

In this study, we present a numerical application to show the effectiveness of the proposed algorithm. The mixing zone which is subjected to high strain and strain rate is located in the neighborhood of the welding tool. That is why only a small and circular region is considered to model the mixing process leading to a reasonable computation time. The proposed algorithm is the first step to validate the material mixing procedure in a bidimensional analysis. It can be extended to a thermomechanical analysis by introducing the energy equation.

In these applications, we consider a circular plate of radius \( r_p = 6 \) mm made of aluminum alloy AA7075 with the viscoplastic material proprieties: \( k = 2.69 \times 10^{10} \) N mm\(^{-2}\), \( A = -3.3155 \), and \( n = 0.1324 \) (see (8)). We perform a time integration by the implicit scheme and we choose \( \Delta t = 0.0001 \) s, \( T = 300 \) K, and \( \delta = 10^{-6} \). The welding tool is considered rigid with radius \( r_t = 1.5 \) mm. In the proposed work, we consider only the dwell phase and the following boundary and initial conditions:

\[
u(x, y, t) = \begin{cases} -r_t \omega \cos(\omega t) & \text{for } x^2 + y^2 = r_t^2 \forall t \geq 0, \\ r_t \omega \sin(\omega t) & \text{for } x^2 + y^2 = r_t^2 \forall t \geq 0, \end{cases}
\]
where $\omega = 100 \text{ rad s}^{-1}$ is the rotation speed. Results of our approach are compared with those obtained using Fluent software. This code is based on an Eulerian formulation and a finite volume discretization. To validate the results of our algorithm, we choose two equivalent configurations between the two formulations update Lagrangian and Eulerian. The CMLS calculation is performed until a time $t = t_c = 0.1 \text{ s}$. In the Eulerian formulation, unsteady calculation with a time $t_c$ is performed. The two calculations use the same constitutive law (8).

5.1. Influence of the Distribution of Points. In this section, to verify the robustness of the proposed numerical algorithm, we propose to study at first some cases of simulation to test the algorithm with different distributions of points in the study domain. These tests allow us to perform a study on the sensitivity to the density of points. We have represented, in Figure 3, the domain configuration for four different distributions of points, where $h = 4dr$ and $dr$ is the interpoint distance. The results shown in these numerical tests shall be registered at time $t_e = 0.1 \text{ s}$.

Figures 4, 5, and 6 show the distribution of the strain rate components $\dot{\epsilon}_{xx}$, $\dot{\epsilon}_{yy}$, and $\dot{\epsilon}_{xy}$ obtained with the CMLS algorithm at different interpoints distances $dr = 5 \cdot 10^{-4} \text{ m}$, $dr = 3 \cdot 10^{-4} \text{ m}$, $dr = 2.5 \cdot 10^{-3} \text{ m}$, and $dr = 2.25 \cdot 10^{-3} \text{ m}$ respectively.

We can conclude that the distribution of points has an influence on the distribution of the components of the strain rate as shown by the results. Analysis of the results shows that the algorithm converges from the interpoint distance $dr = 2.5 \cdot 10^{-4} \text{ m}$, which is equivalent to 1780 points. In the following studies, the equivalent configuration to the interpoint distance $dr = 2.5 \cdot 10^{-4} \text{ m}$ is used in different simulations because this is the optimum choice.

5.2. Influence of the Support Size. The support size must be large enough to cover several collocation points and to avoid numerical drawback and it must be sufficiently small to avoid obtaining very smooth solution. In this simulation, we choose a fixed parameter $h$. In the general case, this parameter should vary during the simulation to be adapted to the change of the point distribution in the domain. In this study, the variation of the interpoints distances is very small justifying the use of $h$ constant.

In this section, the goal is to determine the optimal size of the support (influence domain). For this, we chose a range of values of this size $h$ as shown in Table 1. The results show the existence of an optimum equal to $h = 2.5dr$ for which the maximum relative difference, in comparison with Fluent results, is less than 5%. In the following studies, the optimum choice of support size $h = 2.5dr$ is used in different simulations.

5.3. Influence of the Weighting Function. The CMLS method based on strong formulation forces us to treat higher order derivatives; therefore, it is necessary to use higher order weighting function to ensure the numerical computation of these derivatives. We propose here to study the effect of these weighting functions on the effectiveness of the proposed algorithm. Table 2 shows that the quality of the solution obtained by the algorithm is better in using a weighting
function of an order three (cubic spline) or more wherein the relative difference is less than 5%.

5.4. Validation of the Implicit Algorithm Using the Industrial Code Fluent. In the present section, we propose a comparative and qualitative study of our algorithm with the industrial code Fluent using the optimum choices.

Figure 7 shows the geometry and the space distribution of the material points. Two colors are used to differentiate the two plates to be welded.

Figure 8 presents different configurations of material mixing at different times. One can observe the high deformation level around the welding tool. This result, which is difficult to achieve using finite element method, shows the potential of meshless techniques for modeling FSW process.

The distributions of $u$ and $v$ components of the velocity vector are shown in Figures 9 and 10, respectively. Figures 12 and 13 show the evolution of $u$ and $v$ along the horizontal and vertical sections and the comparison between CMLS and Fluent results (see Figure 11). One can observe that the maximum relative difference is less than 1% confirming the validity of the proposed algorithm.

Figures 14, 15, and 16 show the distribution of the strain rate components ($\dot{\epsilon}_{xx}$, $\dot{\epsilon}_{yy}$, $\dot{\epsilon}_{xy}$) with the comparison between MLS and Fluent results. A maximum relative difference of 5% is obtained.

Figure 15: Distribution of $\dot{\epsilon}_{yy}$ component of the strain rate given by CMLS (a) and Fluent (b).

Figure 16: Distribution of $\dot{\epsilon}_{xy}$ component of the strain rate given by CMLS (a) and Fluent (b).
6. Conclusion

In the present work, we have proposed an implicit iterative algorithm to simulate material mixing observed in friction stir welding which is very difficult to achieve using finite element method. It is based on the combination of continuous moving least square spatial discretization, an Euler implicit scheme time discretization, and an iterative method. The performance of this implicit algorithm has been tested on the viscoplastic problem. Results of the proposed algorithm are compared with those obtained by the industrial code Fluent. The test carried out on the considered example establishes that the proposed algorithm is robust and efficient.

In this work, we considered sticking conditions between tool and plates to be welded. The contact conditions between shoulder and plates are the next challenges to solve before this algorithm can be applied to FSW. We limited ourselves to bidimensional geometry but the work is in progress to extend the proposed algorithm to three-dimensional one with a complex geometry of the welding tool.

References


Submit your manuscripts at http://www.hindawi.com