

An MPP version of the Electromagnetism module in LS-DYNA for 3D Coupled Mechanical-Thermal-Electromagnetic simulations

P. L'Eplattenier¹, C. Ashcraft¹, I. Ulacia²

¹ Livermore Software Technology Corporation, 7374 Las Positas Road, Livermore, CA 94551, USA.

² Mondragon Goi Eskola Politeknikoa, Mondragon Unibertsitatea, Loramendi 4, Mondragon 20500, Spain.

Abstract

LS-DYNA is a general multi-purpose explicit and implicit finite element program used to analyse the non-linear dynamic response of three-dimensional solids and fluids. It is developed by Livermore Software Technology Corporation (LSTC). A new electromagnetism module is being developed in LS-DYNA for coupled mechanical/thermal/electromagnetic simulations. One of the main applications of this module is Electromagnetic Metal Forming (EMF). The electromagnetic fields are solved using a Finite Element Method (FEM) for the conductors coupled with a Boundary Element Method (BEM) for the surrounding air/insulators. Both methods use elements based on discrete differential forms for improved accuracy.

Recently, a Massively Parallel Processing (MPP) version of the EM module was developed allowing sharing the CPU and memory between different processors and thus faster computations on larger problems. The implementation of the FEM and BEM in MPP will be presented.

Finally, the EM module will be illustrated on an actual EMF case. Experimental and numerical results will be compared and the speed-up of the MPP version will be studied.

Keywords

Modelling, Finite Element Method (FEM), Boundary Element Method (BEM)

1 Introduction

LS-DYNA is a general-purpose nonlinear finite element program that is capable of simulating complex real world problems [1]. The major development goal of Livermore Software Technology Corporation (LSTC) is to provide within LS-DYNA capabilities to seamlessly solve problems that require multi-physics, multiple-stages, and multi-

processing. LS-DYNA is suitable to investigate phenomena that involve large deformations, sophisticated material models and complex contact conditions. LS-DYNA allows running an analysis explicitly or implicitly and combining different disciplines such as coupled thermal analysis, fluid dynamics, fluid-structure interaction, SPH (Smooth Particle Hydrodynamics), EFG (Element Free Galerkin). Metal forming is one of LS-DYNA's main applications, with capabilities to simulate rolling, extrusion, forging, casting, spinning, ironing, super-plastic forming, sheet metal stamping, profile rolling, deep drawing, hydro-forming, multi-stage processing, springback, hemming [1].

An electromagnetism (EM) module is under development in LS-DYNA in order to perform coupled mechanical/thermal/electromagnetic simulations [2],[3],[4]. Electromagnetic Metal Forming (EMF) is the main application of this development, but other processes could also be simulated, where magnetic pressure induces mechanical stress and deformations and/or the Joule effect induces a heating process: magnetic metal cutting, magnetic metal welding, very high magnetic pressure generation, computation of the stresses and deformations in various coils, magnetic flux compression, induced heating and so forth. This module allows us to introduce some source electrical currents into solid conductors, and to compute the associated magnetic field, electric field, as well as induced currents. These fields are computed by solving the Maxwell equations in the eddy-current approximation. The Maxwell equations are solved using a Finite Element Method (FEM) [5] for the solid conductors coupled with a Boundary Element Method (BEM) [6] for the surrounding air (or insulators). Both the FEM and the BEM are based on discrete differential forms (Nedelec-like elements [7]).

The computation of the electromagnetic fields, and especially the BEM part, is very time consuming and requires a lot of memory. An MPP implementation [8],[9] allows reducing both the CPU time and the memory requirement by splitting them between different processors. An MPP version of the mechanical and thermal modules already exists in LS-DYNA. More recently, a Massively Parallel Processing (MPP) version of the EM module has been introduced, allowing coupled mechanical-thermal-electromagnetic simulations in MPP.

This paper is organized as follows: In part 2, we will present the physics of the EM part of the problem, its coupled finite element / boundary elements representation, and the respective solvers.

In part 3, the MPP implementation of the EM module will be presented, and in part 4, the MPP version will be illustrated on an EMF case. Numerical and experimental results will be compared and the speed-up of the MPP version will be studied.

2 Presentation of the model

In this section, we will present only the EM part of the model. For details on the Mechanical and Thermal parts, the reader should consult [1].

2.1 Presentation of the physics

Let Ω be a set of multiply connected conducting regions. The surrounding insulator exterior regions will be called Ω_e . The boundary between Ω and Ω_e is called Γ , and the (artificial) boundary on Ω at the end of the meshing region (hence where the conductors are connected to an external circuit) is called Γ_c . In the following, we will denote \vec{n} as the

outward normal to surfaces Γ or Γ_c . The electrical conductivity, permeability and permittivity are called σ , μ and ε respectively. In Ω_e , we have $\sigma = 0$ and $\mu = \mu_0$.

We solve the Maxwell equations in the so called low frequency or “eddy-current” approximation, which is valid for good enough conductors with low frequency varying fields such that the condition $\varepsilon \frac{\partial \vec{E}}{\partial t} \ll \sigma \vec{E}$, where \vec{E} is the electric field, is satisfied. When using a vector potential \vec{A} and scalar potential ϕ representation and using the Gauge condition $\vec{E} = -\vec{\nabla}\phi - \frac{\partial \vec{A}}{\partial t}$, we end up with the following system to solve [4],[10]:

$$\nabla \cdot \sigma \vec{\nabla} \phi = 0 \quad (1)$$

and

$$\sigma \frac{\partial \vec{A}}{\partial t} + \vec{\nabla} \times \frac{1}{\mu} \vec{\nabla} \times \vec{A} + \sigma \vec{\nabla} \phi = \vec{j}_s \quad (2)$$

Where \vec{j}_s is a divergence free source current density.

2.2 The coupled Finite Element Method/Boundary Element Method

We use Nedelec “edge elements” [7] and call $w^0, \vec{w}^1, \vec{w}^2$, and w^3 the basis functions associated respectively with the 0, 1, 2, and 3-forms [11]. Equation (1) is projected against 0-forms basis functions and equation (2) against 1-forms to give, after using the appropriate Greens vector identities [11],[12],[13]:

$$\int_{\Omega} \sigma \vec{\nabla} \phi \cdot \vec{\nabla} w^0 d\Omega = 0 \quad (3)$$

$$\int_{\Omega} \sigma \frac{\partial \vec{A}}{\partial t} \cdot \vec{w}^1 d\Omega + \int_{\Omega} \frac{1}{\mu} \vec{\nabla} \times \vec{A} \cdot \vec{\nabla} \times \vec{w}^1 d\Omega = - \int_{\Omega} \sigma \vec{\nabla} \phi \cdot \vec{w}^1 d\Omega + \frac{1}{\mu} \int_{\Gamma} [\vec{n} \times (\vec{\nabla} \times \vec{A})] \cdot \vec{w}^1 d\Gamma \quad (4)$$

or equivalently after decomposing \vec{A} and ϕ respectively on the 0-form and 1-form basis functions [4]:

$$S^0(\sigma)\phi = 0 \quad (5)$$

$$M^1(\sigma) \frac{da}{dt} + S^1\left(\frac{1}{\mu}\right)a = -D^{01}(\sigma)\phi + Sa \quad (6)$$

Where we introduced the 0-form stiffness matrix S^0 , the 1-form mass matrix M^1 , the 1-form stiffness matrix S^1 and the 0-1 form derivative matrix D^{01} [13]. The last term of equation (6) which involves the “outside matrix stiffness” S is computed using a BEM [6]:

An intermediate variable “surface current” \vec{k} is introduced. This surface current, defined on the boundary Γ is such that it produces the same vector potential (and thus \vec{B} field) in the exterior regions Ω_e as the actual volume current flowing through the conductors [4],[14]:

$$\vec{A}(x) = \frac{\mu_0}{4\pi} \int_{\Gamma} \frac{1}{|x-y|} \vec{k}(y) dy \quad \text{for all } x \in \Omega_e \quad (\text{and in particular for all } x \in \Gamma). \quad (7)$$

One then has:

$$[\vec{n} \times (\vec{\nabla} \times \vec{A})](x) = \frac{\mu_0}{2} \vec{k}(x) - \frac{\mu_0}{4\pi} \int_{\Gamma} \frac{1}{|x-y|^3} \vec{n} \times [(\vec{x}-\vec{y}) \times \vec{k}(y)] dy$$

$$\text{for } x \rightarrow x_0 \in \Gamma \quad (8)$$

When projecting these equations on the 1-forms basis functions for \bar{A} and the “twisted” 1-forms $\bar{v}^1(x) = \bar{n} \times \bar{w}^1(x)$ for \bar{k} one gets the following matrix equations [4]:

$$Pk = Da \quad (9)$$

$$Sa = Qk \equiv Q_s k + Q_d k \quad (10)$$

Where we introduced the BEM matrices

$$P_{i,j} = \frac{\mu_0}{4\pi} \iint_{\Gamma_x \Gamma_y} \frac{1}{|x-y|} \bar{v}_i^1(x) \bullet \bar{v}_j^1(y) d\Gamma_x d\Gamma_y, \quad D_{i,j} = \int_{\Gamma_x} \bar{v}_i^1(x) \bullet \bar{w}_j^1(x) d\Gamma_x \quad (11)$$

$$Q_{S_{i,j}} = \frac{1}{2} \int_{\Gamma_x} \bar{w}_i^1(x) \bullet \bar{v}_j^1(x) d\Gamma_x, \quad (12)$$

$$Q_{D_{i,j}} = -\frac{1}{4\pi} \iint_{\Gamma_x \Gamma_y} \frac{1}{|x-y|^3} \bar{w}_i^1(x) \bullet \{ \bar{n}_x \times [(\bar{x} - \bar{y}) \times \bar{v}_j^1(y)] \} d\Gamma_x d\Gamma_y \quad (13)$$

2.3 Domain decomposition

Contrarily to the FEM matrices, the BEM matrices P and Q_d are full dense and cannot be stored as dense arrays since the memory requirement would grow up very quickly with the size of the system. In order to limit the memory requirement, a domain decomposition is done on the BEM mesh, which splits the BEM matrices into submatrices. On the off-diagonal submatrices, a low rank approximation based on a rank revealing QR decomposition is performed [15],[16]. For submatrices corresponding to far away domains, the rank can be significantly smaller than the size of the submatrix, thus reducing the storage of the submatrix. We typically see reductions of by factors around 20 between the full dense matrix and the block matrix with low rank approximations. This low rank approximation also speeds up the matrix * vector operation used intensively in the iterative method to solve the BEM system (equation 9).

2.4 Solvers

The coupled FEM/BEM system is solved in an iterative way [4]:

$$Pk_{n+1}^{t+1} = Da_n^{t+1} \quad (14)$$

$$[M^1(\sigma) + dtS^1(\frac{1}{\mu})]a_{n+1}^{t+1} = M^1(\sigma)a^t - dtD^{01}(\sigma)\phi^{t+1} + dtQk_{n+1}^{t+1} \quad (15)$$

until convergence on both k_n^{t+1} and a_n^{t+1} .

The dense system (14) is solved using a pre-conditioned gradient (PCG) method [15] where different preconditioners [17] can be used, such as a diagonal line or a diagonal block preconditioner. Other types of preconditioners are being developed. When successive systems have to be solved with the same matrix and different right hand sides (rhs), as it happens in time domain problems, the solution for a given rhs can be used as an initial solution for the next system. This is done when the successive rhs's are nearly parallel and the initial solution is then adjusted by the ratio of the norms of the two rhs. Starting with a good initial guess considerably reduces the number of iterations in the subsequent solves.

The sparse system (15) is solved using a direct solver [1].

3 MPP implementation

3.1 Introduction

The assembly and solve of the electromagnetic systems, and especially the BEM one which involves dense matrices are computationally expensive. The use of an MPP architecture [15],[16] allows sharing some of the computations between the different processors. It also allows sharing the storage of the different matrices between the processors, and thus gives an overall gain in memory and in computational time. We will now present how the electromagnetism systems are implemented in MPP. We will insist on the BEM implementation where the gains are maximal. In the following, we will call n_p the number of processor for a given simulation.

3.2 Implementation of the FEM part in MPP

The FEM part of the EM systems is handled in the same manner as the mechanical and thermal systems. For details, the reader should consult [1].

A domain decomposition with n_p domains is defined on the solid mesh. This is a serial operation performed on one processor. Each processor is assigned one domain. Associated with this domain is mesh data (node position and element connectivity) which largely stays local to each processor.

An FEM matrix is computed in parallel. Each processor creates elemental matrices for the elements it owns and the partial matrices on the processors are assembled and partitioned (an “all-sum” operation). The matrix-vector multiplies become sequences of serial computations followed by collective “all-sum” and “all-gather” operations.

3.3 Implementation of the BEM part in MPP

The BEM system deals with dense matrices and thus requires a lot of computational time, both for the matrices assembly and for solving the systems. A large gain can be made by using an MPP implementation. Unlike the FEM system where only one element is needed to compute an elemental matrix, the elemental BEM matrices involve double integrals over 2 elements, like in equations (11) and (13). The full BEM mesh (node position, BEM faces connectivity) is thus broadcasted to each processor.

3.3.1 BEM matrices: singular integration

The entries of a typical BEM matrix P are computed by integrating a kernel, smoothed by the basis functions between 2 faces, like in equations (11) or (13). Since the kernel is a negative power of the distance between the faces, these integrals can become singular or near singular for self or neighbour faces. Such singular integrations are based on Duffy transform which regularizes the kernel [18],[19]. The transforms depend on the type of face (quadrilateral or triangular) and on the type of singularity (self face, faces with common edges, faces with common nodes and so forth).

They require many integration points on each face and the evaluation of the kernel between these integration points and are thus numerically expensive.

In MPP, the pairs of neighbour BEM faces are classified based on a cut-off on the face separation parameter $p = \frac{\text{dist}(C_1, C_2)}{\max(R_1, R_2)}$

Where C_1 and C_2 are the centers of faces 1 and 2 respectively, and R_1 and R_2 their respective radii. The smaller the value of p , the more singular the pair (face 1, face 2) is and the more integration points are needed. The neighbour pairs are distributed among the processors to balance the singular integration work to compute P_s , the singular part of P .

3.3.2 BEM matrices: regular integration

A domain decomposition is performed on the BEM mesh with n_d domains ($n_d \succ n_p$). This domain decomposition allows decomposing any BEM matrix P as a block matrix:

$$P = \sum_{\Omega_1, \Omega_2 \in \delta} P(\Omega_1, \Omega_2) \quad (16)$$

Where $P(\Omega_1, \Omega_2)$ represents the submatrix of matrix P with the rows in Ω_1 and the columns in Ω_2 .

For any 2 domains Ω_1 and Ω_2 we call the separation parameter:

$$p(\Omega_1, \Omega_2) = \frac{\text{dist}(C_1, C_2)}{\max(R_1, R_2)} \quad (17)$$

Where C_1 and C_2 are the centers of domains 1 and 2 respectively, and R_1 and R_2 their respective radii.

The n_d domains are partitioned between the n_p processors using the following method:

- n_p “center” domains are chosen so that they are as well separated as possible, i.e. so that their separation parameters (17) are maximized.
- Each center domain is assigned to a processor.
- The other domains are assigned to the processor whose center domain is closest, by keeping the number of domain per processor as uniform as possible.

We thus end up with a domain partition such that each processor owns its domain center plus all the surrounding domains. An example of such a domain partition is presented on figure 1.

The sum (16) is partitioned between the processors using the following method:

- $P(\Omega_1, \Omega_2)$ is computed either by the processor which owns Ω_1 or by the processor which owns Ω_2 .
- The couples of domain are split between the processors in order to keep the same number of couples per processor.

The singular part P_s is then block partitioned with the same partition (16) used for the regular part of the matrix, and each singular block is added to the corresponding regular block in order to assemble the full matrix P .

All together, the amount of work to assemble the matrix is thus evenly split between the processors, as well as the memory needed to store the matrix.

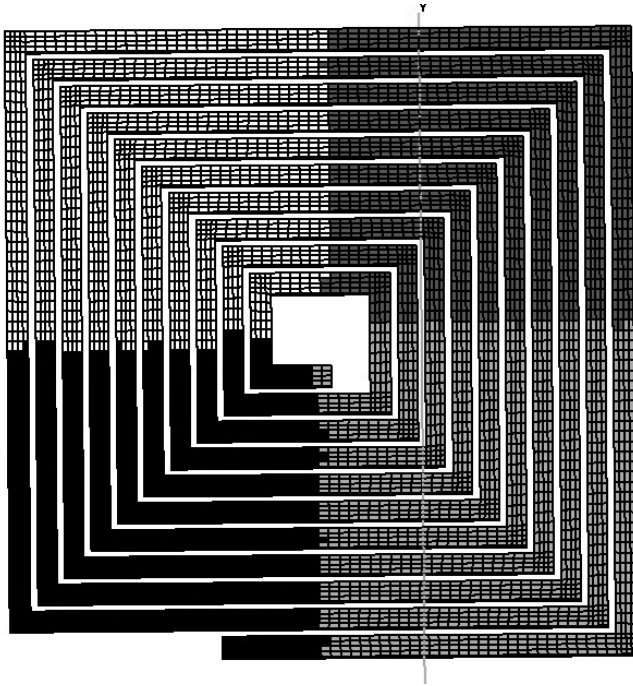


Figure 1: Example of a domain partition of a BEM mesh into 4 processors

4 Results

4.1 Presentation of the EMF experiments and numerical models

The material used in this experiment was a commercial AZ31B magnesium alloy sheet for which the mechanical and microstructural evolution at high strain rate can be found in [20]. A square free forming operation is considered for the numerical study: The employed coil and die are shown in Figure 2. More details about the experiments can also be found in [21]. The selection of these operations is motivated by the fact that the deformation of the workpiece could be recorded by a high speed camera in the experiments and thus not only the final results but also their evolution in time can be compared.

A Photron FASTCAM-APXTM high speed camera was used to record the deformation at a sampling rate of 37500 fps. Moreover, in free forming operations, the influence of the die geometry in the final shape of the workpiece is less noticeable than in a close die forming operation and therefore it makes the numerical simulation more challenging, evidencing possible discrepancies with experiments.

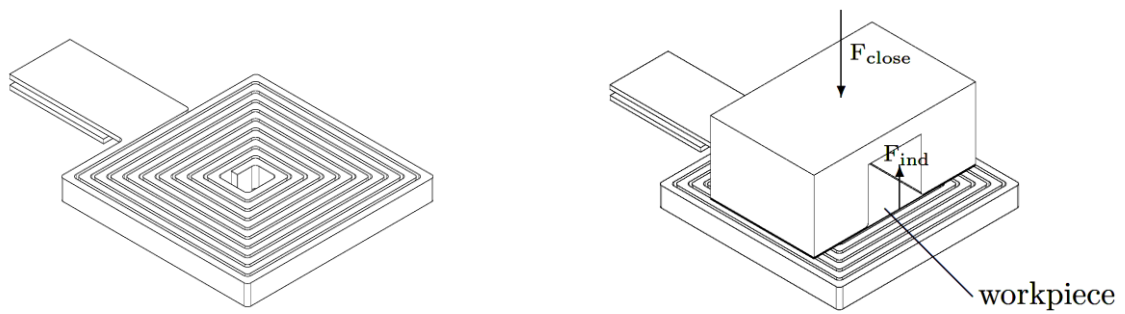


Figure 2: Coil and die used for the square free forming case.

The resistance, inductance and capacitance values of the equivalent RLC circuit used in the experiments are respectively $R_0=0.956$ m Ω , $L_0=10$ nH and $C_0=180$ μ F. A charging voltage of 5.773 kV (i.e. 3 kJ) is used in the numerical simulations.

The clamping force is applied as nodal forces between the die and the blank holder.

The electromagnetic analysis is performed up to 300 μ s with a time step of 1.5 μ s and a recomputation of the BEM and FEM matrices every 10 steps. There thus are 200 electromagnetic time steps and 20 matrices reassembly in each simulation. The thermo-mechanical analysis is performed up to 300 μ s with $t_{\text{therm}}=2$ μ s and $t_{\text{mech}}=30$ ns respectively.

The main electrical parameter for non magnetic materials, such as non-ferrous materials like copper or magnesium alloys, is the electrical conductivity (σ_e). In the numerical models the temperature dependence of the electrical conductivity is taken into account whereas the influence of the plastic deformation is not considered. The temperature dependence of the electrical conductivity is modelled using a Burgess model [22] and also using linear interpolation between conductivity values at given temperatures. The evolution of the electrical conductivity of copper (i.e. the coil) with temperature is taken from [22] and the conductivity values for the AZ31B magnesium alloy (i.e. the workpiece) are taken from [23]. The die and the binder are considered insulators, so there is not current calculation in these bodies since the skin depth is usually small in EMF operations.

The die and the blank-holder are considered as rigid bodies made of steel ($\rho = 7.8$ g/cm³; $E = 210$ GPa; $\nu = 0.33$) in order to save computational time. The coil is modelled as an elastic material of very high strength to avoid any deformation, since the study is not focused in the coil but in the deformation behaviour of the workpiece. The assumption of the coil to work in the elastic domain is faithful when the resin in which it is embedded resists the reaction forces produced by the EMF discharges. In order to model the constitutive response of the workpiece, the flow stress is determined as a function of the plastic strain, the strain rate and the temperature using the constitutive equation given by Johnson and Cook [24]. The values employed in this paper are taken from [25].

The thermal properties considered in this study are the specific heat capacity (c_p), the thermal conductivity (κ) and the coefficient of thermal expansion (α). Data for oxygen free copper is taken from ASM Specialty Handbook for Copper and Copper alloys [26] and data for AZ31B magnesium alloy is taken from ASM Specialty Handbook for Magnesium and Magnesium alloys [23].

For the numerical study, different cases have been studied, with different coil mesh densities representing the same geometry. The model corresponding to Mesh1 is shown in Figure 3 and table 1 gives the statistics of the different meshes. All the cases have been solved with the same tolerances in the PCG method and with the same block-diagonal pre-conditioner.

Case name	Mesh1	Mesh8	Mesh16	Mesh32	Mesh64
# nodes	39504	63759	86537	131747	420478
# elements	28595	43568	62600	98744	343224
#BEM Pdofs	19851	36971	44279	62363	147843
#BEM Qdofs	39688	73920	88532	124688	295648

Table 1: Statistics on the different meshes used for the numerical study.

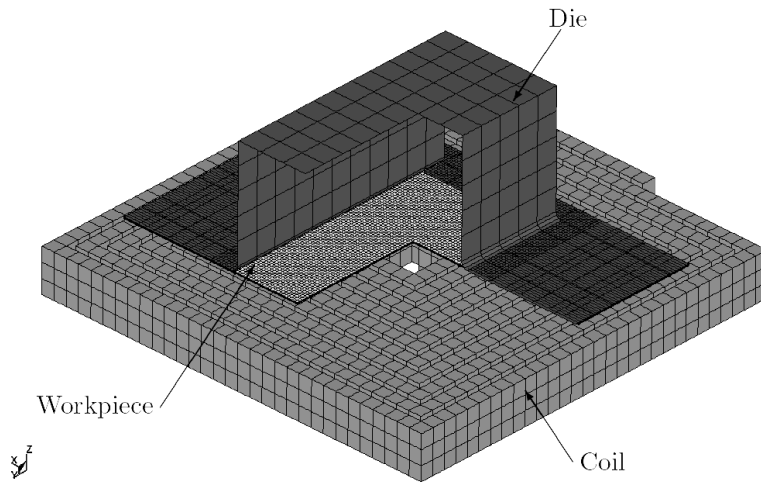


Figure 3: Model corresponding to Mesh1 (a quarter of the workpiece and die is removed to show the coil).

4.2 Numerical-experimental comparisons

The experimental and numerical final shapes of the workpiece showed a good agreement as pictured on figure 4. The numerical and experimental Z displacement versus time were also compared and showed a very good agreement as shown on figure 5. More details can be found in [21]. The numerical results given by the different meshes are fairly consistent with of course more details on the thinner meshes.

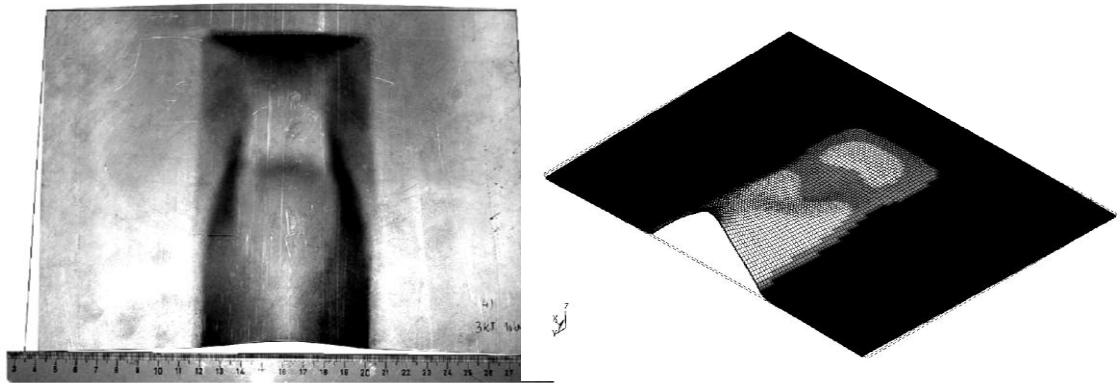


Figure 4: Final shape of the workpiece: experimental (left) and numerical (right).

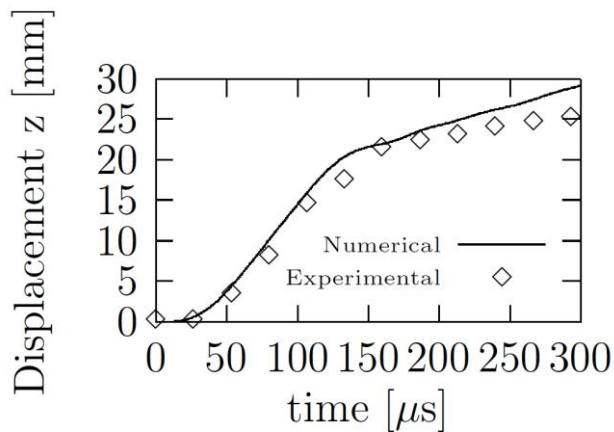


Figure 5: Comparison between numerically predicted and experimentally observed time evolution of the maximum height of the workpiece.

4.3 Study of the MPP speed-up

The different cases have been simulated using 1, 2, 4, 8, 16, 32, or 64 processors. Table 2 shows the total computational time versus the number of processors for the different cases and figure 6 shows the computational time vs the number of processors for mesh1 and mesh8. One can notice the large reduction in computational time for Mesh1 when going from 1 to 2 and then to 4 processors. This reduction is less dramatic when increasing even more the number of processor on such a small case. This is because for large numbers of processors, the communication between processors becomes very important compared to the work done by each processor. We looked in more detail at the two most time consuming parts of the computation, i.e. the BEM matrices assembly, which we will call “assembly” and the BEM solve using a PCG method which we will call “solve”. On both mesh1 and mesh8, the assembly time scales pretty well with the number of processors. The solve time does not scale as well, but represents a small fraction of the total time on these small cases.

The larger test cases, mesh16, mesh32 and mesh64 can only be run on larger number of processors due to the memory requirement. On these cases, doubling the number of processors shows a significant reduction in the computational time.

# processors	Mesh1	Mesh8	Mesh16	Mesh32	Mesh64
1	24:16	55:41			
2	12:16	27:42			
4	6:28	14:13			
8	3:56	8:49	12:14		
16	2:29	5:15	7:34	15:53	
32	1:36	3:07	4:22	8:57	42:16
64	1:09	2:01	2:46	5:32	26:31

Table 2: Computational time (hours:minutes) versus number of processors for the different meshes.

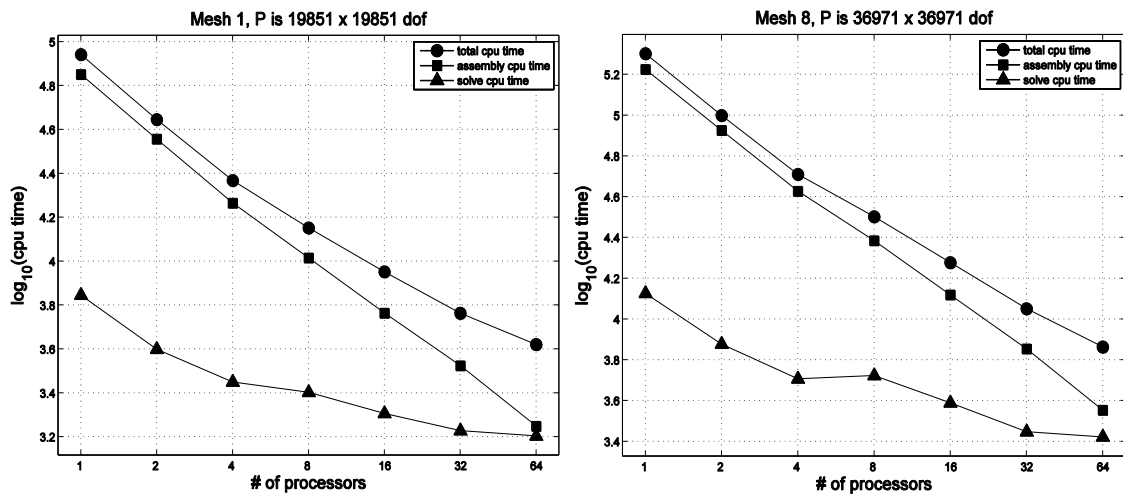


Figure 6: Computational time (log scale) versus number of processors for Mesh1 (left) and mesh8 (right). The dots represent the total time, the squares the assembly time, and the triangles the solve time.

In order to better compare the speed-up, we define the efficiency of a simulation performed with p processors compared to a reference simulation performed with p_0 processors ($p_0 \leq p$ and $p_0 = 1$ whenever the total memory needed by the run makes it possible) as:

$$E_p = \frac{p_0 T_{p_0}}{p T_p}, \tag{18}$$

Where T_{p_0} is the computational time for p_0 processor, and T_p is the computational time for p processors. Ideally, we would like this efficiency to be equal to 1, and lower

values show time spent (“wasted”) in communication and synchronization. Figure 7 shows the total computational time versus the number of processors for all cases, as well as the efficiency versus the number of processors for all the cases. One can see that for the small cases, the efficiency falls down below 0.5 for large number of processors, but for the larger cases, it stays above 0.6 even at large numbers of processors. In the future, better pre-conditioners will be introduced in order to improve the solve efficiency and thus the overall efficiency.

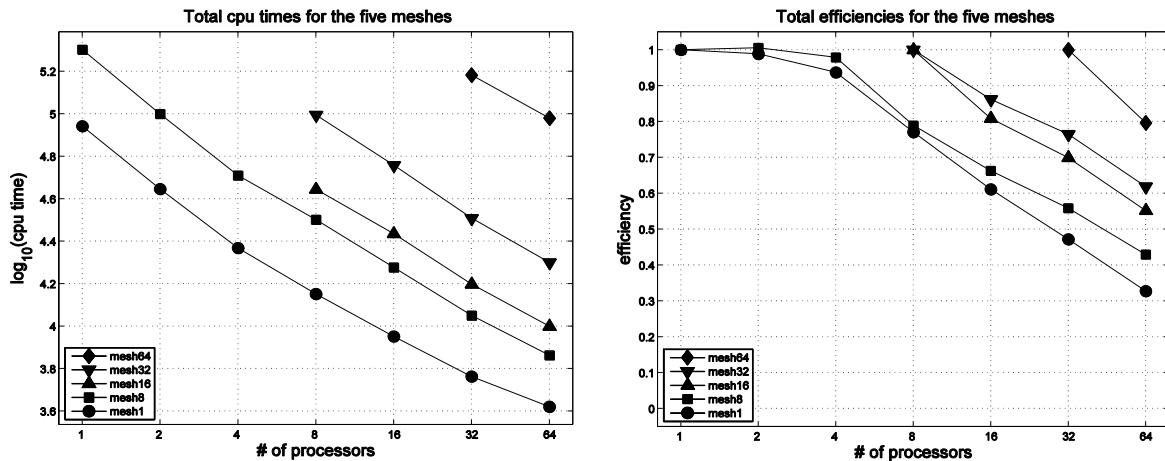


Figure 7: Total CPU time (left) and total efficiency (right) versus number of processors for the different cases: dots: mesh1, squares: mesh8, up triangles: mesh16, down triangles: mesh32, diamonds: mesh64.

5 Conclusions

The newly introduced Electromagnetism module of LS-DYNA was presented. The electromagnetic fields are computed by solving the Maxwell equations in the eddy-current approximation, using a FEM for the conductors coupled with a BEM for the surrounding air and insulators. The implementation of the MPP version of the module was presented, and in particular, the BEM system part. Results were presented on an EMF test case, and the speed-up of the MPP version was presented. The MPP version allows much faster simulation and much larger cases due to the distribution of the memory between the different processors.

Further improvements on the speed-up are currently being addressed, such as the introduction of new pre-conditioners in MPP which could considerably improve the efficiency of the solve part.

More recent developments in the EM module include an induced heating and resistive heating capability, as well as a sliding contact capability for the electromagnetic fields which allows the simulation of rail gun like experiments. In the future, the EM module will be extended from hexahedral to tetrahedral and wedge solid elements as well as shell elements, and a mesh adaptivity capability will be introduced. Finally, other solvers than the eddy-current will be introduced, including a magnetostatics solver.

The EM module is integrated in the 980 version of LS-DYNA, which should be released sometime in 2010. In the mean time, it is available as a “beta version”.

References

- [1] LS-DYNA Theory Manual, LSTC.
- [2] *P. L'Eplattenier, G. Cook, C. Ashcraft, M. Burger, A. Shapiro, G. Daehn, M. Seith*, "Introduction of an Electromagnetism Module in LS-DYNA for Coupled Mechanical-Thermal-Electromagnetic Simulations", 9th International LS-DYNA Users conference", Dearborn, Michigan, June 2005.
- [3] *P. L'Eplattenier, G. Cook, C. Ashcraft*, "Introduction of an Electromagnetism Module in LS-DYNA for Coupled Mechanical-Thermal-Electromagnetic Simulations", Internatinal Conference On High Speed Forming 08, March 11-12, 2008, Dortmund, Germany.
- [4] *P. L'Eplattenier, G. Cook, C. Ashcraft, M Burger, J. Imbert, M. Worswick*, "Introduction of an Electromagnetism Module in LS-DYNA for Coupled Mechanical-Thermal-Electromagnetic Simulations", Steel Research Int. 80 (2009) no 5.
- [5] *J. Jin*, "The Finite Element Method In Electromagnetics", Wiley, 1993.
- [6] *J. Shen*, "Computational Electromagnetics Using Boundary Elements, Advances In Modelling Eddy Currents", Topics in Engineering Vol 24, Series Eds: C.A. Brebbia and J.J. Connor, Southampton and Boston: Computational Mechanics Publications, 1995".
- [7] *J.C. Nedelec*, "A New Family of Mixed Finite Elements in R³", Num. Math, 50:57-81, 1986.
- [8] *Marc Snir, Steve Otto, Steven Huss-Lederman, David Walker, Jack Dongarra*, "MPI, the complete reference, Volume I, the MPI core", Scientific and Engineering Computation Series, MIT press, 1998.
- [9] *William Gropp, Ewing Lusk, Anthony Skjellum*, "Using MPI, Portable Parallel Programming with the Message-Passing Interface", Second Edition, Scientific and Engineering Computation Series, MIT press, 1998.
- [10] *O. Biro and K. Preis*, "On the use of the magnetic vector potential in the finite element analysis of three-dimentional eddy currents", IEEE Transaction on Magnetics, 25(4)3145-3159,1989.
- [11] *P. Castillo, R. Rieben and D. White*, "FEMSTER: An object oriented class library of discrete differential forms". In Proceedings of the 2003 IEEE International Antennas and Propagation Symposium, volume 2, pages 181-184, Columbus, Ohio, June 2003.
- [12] *R. Rieben*, "A Novel High Order Time Domain Vector Finite Element Method for the Simulation of Electromagnetic Devices", Ph-D Thesis, University of California Davis, 2004.
- [13] *R. Rieben and D. White*, "Verification of high-order mixed finite element solution of transient magnetic diffusion problems", IEEE Transaction on Magnetics, 42(1), 25-39, 2006.
- [14] *Z. Ren, A. Razek*, "New technique for solving three-dimentional multiply connected eddy-current problems", IEE Proceedings, Vol. 137, Pt. A, No 3, May 1990.
- [15] *G. H. Golub and C. F. van Loan*, "Matrix Computations", Third Edition, Johns Hopkins University Press, Baltimore, MD, 1996.
- [16] *P. A. Businger and G. H. Golub*, "Linear least squares solution via Householder transformations", Numer. Math, volume 7, 269-276, 1965.

- [17] *K.E. Chen*, "On a class of preconditioning methods for dense linear systems from boundary elements", *SIAM J. Sci. Comput.* 20(2), 684-698, 1998.
- [18] *D.J. Taylor*, "Accurate and Efficient Numerical Integration of Weakly Singular Integrals in Galerkin EFIE Solutions", *IEEE Transactions on Antennas and Propagation*, vol. 51, no 7, July 2003.
- [19] *A. Frangi, G.Novati, R.Springhetti, and M.Rovizzi*, "3D fracture analysis by the symmetric Galerkin BEM", *Computational Mechanics* 28 (2002).
- [20] *I. Ulacia, N.V. Dudamell, F. Gálvez, S. Yi, M.T. Pérez-Prado, I. Hurtado*, "Mechanical behavior and microstructural evolution of a Mg AZ31 sheet at dynamic strain rates", *Acta Mater*, 2010, doi:10.1016/j.actamat.2010.01.029.
- [21] *I. Ulacia, I. Hurtado, J. Imbert, C. Salisbury, M. Worswick, A. Arroyo*, "Experimental and numerical study of electromagnetic forming of AZ31B magnesium alloy sheet", *Steel Res. Int.* 80, 2009, p. 344 - 350.
- [22] *T. Burgess*, "Electrical resistivity model of metals", *Proc. 4th Int. Conf. on Megagauss Magnetic Field Generation and Related Topics*, 1986, pp. 307 - 316.
- [23] *M. Avedesian, H. Baker*, "Magnesium and Magnesium Alloys", *ASM Specialty Handbook*. ASM Int., Materials Park, OH, 1999.
- [24] *G. Johnson, W. Cook*, "A constitutive model and data for metals subjected to large strains, high strain rates and high temperatures", *Proc. 7th Int. Symp. on ballistics*. pp. 541 – 547, 1983.
- [25] *I. Ulacia, C.P. Salisbury, I. Hurtado, M.J. Worswick*, "Tensile characterization and constitutive modeling of AZ31B magnesium alloy sheet over a wide range of strain rates and temperatures", *J. Mater. Proc. Tech.* Manuscript submitted for publication, 2009.
- [26] *J. Davis*, "Copper and Copper Alloys", *ASM Specialty Handbook*. ASM Int., Materials Park, OH, 2001.