;

Proceedings in Manufacturing Systems, Volume 13, Issue 1, 2018, 39-46 ISSN 2067-9238

DYNAMIC SIMULATION OF INDUSTRIAL ROBOTS BEHAVIOUR USING INTEGRATED SOFTWARE PROCEDURES

Mihai BOLDA[1], Cristina PUPĂZĂ[2,*]

[1)] PhD Student, Machines and Manufacturing Systems Department, Politehnica University of Bucharest, Romania
[2)] Prof. Dr.-Ing., Machines and Manufacturing Systems Department, Politehnica University of Bucharest, Romania

Abstract: *The paper presents a research regarding the dynamic behavior of an industrial robot during operation. The main target of the attempt was to primarily test a set of software applications in a combination used in the automotive industry: ANSA-Pam-Crash-METAPost. The dynamic analysis has been performed using an explicit solver, which involves time integration and does not require an iterative computation. The robot has been programmed to manipulate a mass of 69 Kg over a distance of 1.5m at a rotational speed of 110˚/s. A comparison was also done between the robot's behavior with and without the manipulated object for the same cycling. The original research was focused on new kinematic modelling and simulation functionalities. It was further continued with an endurance test to provide information on areas that are fatigue-sensitive. The analysis proved that CAE integrated software used mainly in the automotive industry is also a valuable design and research tool for industrial robots when adapting and customizing the simulation methodology.*

Key words: *CAE, industrial robots, simulation, kinematic, transient, fatigue.*

1. INTRODUCTION

Kinematic calculations and Computer Aided Engineering techniques are largely accepted as powerful tools in the design and research of industrial robots. To meet their ever-growing demands, it is of critical importance to understand the robot damage mechanisms for their performance [1]. Although recent literature revealed the importance of the topic only some structural elements of the robot were analyzed from this point of view [2]. A more general approach of fatigue life improvement was focused on the influence of the joint flexibility on the robot workspace [3], which presumes a demanding scientific assessment.

This paper contains a study on the robot computational model from a CAD-FEM perspective with detailed explanation on the robot modelling attempts, followed by an explicit dynamic simulation that takes into account severe operating scenarios. The research was carried out using a combination of integrated software devoted to the automotive industry, but taking into account modelling peculiarities for industrial robots. Areas with potential fatigue vulnerability were identified for large and heavy duty cycles. Because of the complexity of the kinematic configuration of the robot only three numerically controlled axes were considered active. Fatigue limit or endurance limit and fatigue strength are used to describe a property of materials, namely the amplitude of cyclic stress that can be applied to the material without causing fatigue failure [4].

* Corresponding author: Politehnica University of Bucharest, 313 Splaiul Independentei, Sector 6, Bucharest, Romania;
Tel.: 400214029269;
E-mail addresses: *cristinapupaza@yahoo.co.uk*, (C. Pupăză), *bolda.mihaii@gmail.com* (M. Bolda).

The simulation model was inspired by an IRB 760 articulated arm robot manufactured by ABB, but the geometry of the model was entirely reconstructed. The robot has four degrees of freedom.

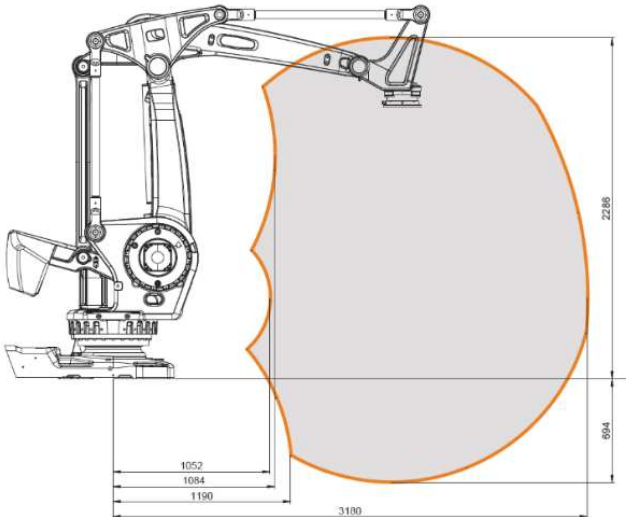

Fig. 1. Articulated arm robot ABB IRB760.

Table 1

Technical specifications of ABB IRB 760

Reach [m]	3.18
Handling capacity [kg]	450
Number of axis	4
Protection	IP67
Mounting	Floor
Controller	IRC5
Base [mm]	1140×800
Weight [kg]	2310
Repeatability [mm]	0.05

2. FEM PROCEDURES

FEM is a general numerical method used to solve differential equations with partial derivatives that describe physical phenomena. The algorithm mainly consists in the decomposition of the analysis domain into simple geometric entities that are analyzed and then the entire domain is recomposed, while respecting certain mathematical requirements.

The partial derivative equations approximate the physical system which has an infinite number of degrees of freedom. Following FEM [5], the partial derivative equations are reduced to systems of algebraic equations describing a discrete system with a finite number of degrees of freedom.

Finite elements occur in the process of dividing the domain of the structure, regardless of the chosen analysis type. Some types of computational models are presented on Fig. 2:

a) one-dimensional elements used for beam structures;

b) two-dimensional elements – triangles and quadrilaterals used in tiles and membrane structures and;

c) tetrahedral and hexahedral elements used for massive structures.

In a broad sense, the finite element wireframe approximates the following properties of the model:

- geometric or model shape, characterized as part of a body having certain dimensions;
- physical, the finite element has attached physical properties, such as: elasticity, density, damping etc.;
- functional: one or more variables of the problem according to nodal values and corresponding functions of the problem.

Depending on the number of degrees of freedom per node, the following categories of finite elements can be distinguished:

- with a single degree of freedom per node, in the case of uniaxial loading;
- with two degrees of freedom per node, for plane loading, for articulated beams, lattice beams, frames and planar spaces;
- with three degrees of freedom.

Geometrical approximations of finite elements are controlled by the number of nodes used outside the element to define the shape, while physical approximations are controlled by the number of nodes

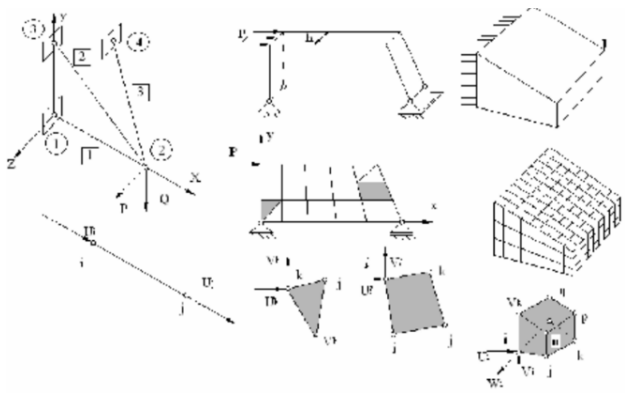

Fig. 2. Types of Finite Element models [5].

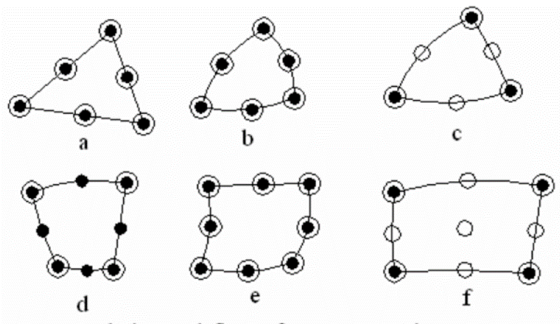

Fig. 3. Different finite element types [5]:
• – nodes that define the geometry;
○ – nodes that approximate state variables.

inside and outside of the element, using different interpolation functions [5]. Depending on physical and geometric approximations, there are three categories of finite elements: subparametric $m < n$ (Fig. 3,*c* and *f*), where m is the degree of the interpolation function N_i and n is the degree of the derivative N_i'; isoperimetric, when $m = n$ (Fig. 3,*b* and *e*) – at the same nodal points the same functions are used to define the geometry and the state variables of the finite element; and super parametric, when $m > n$ (Fig. 3,*a* and *d*).

Determining the shape and displacement of the finite element, the nodes can be represented by interpolation functions in relation to global coordinates.

The conditions to be met by the approximation functions (if derived up to the $k + 1$ order) to ensure the convergence of the finite elements decrease are:

- continuity, which is ensuring small variations of the unknown parameter across the entire range of the finite element, including its boundary. For example, if the finite element has only displacements in the nodes, the approximation function must be continuous, class C0 – Lagrange generalized function, and if the finite element has both nodes and rotations in the nodes, it is necessary that the function approximation and its first derivative to be continuous, class C1 – generalized Hermite type function;
- compatibility – meaning that during the change of the domain of the problem along the common border, the finite elements do not have to separate, and the values of the approximation functions on the common border depend only on the unknowns in the nodes on this border;
- completeness, which is ensured by the way the approximation functions are chosen. For example, in the mechanics of the deformable solid, the approximation function that satisfies the completeness conditions contains modes of displacements that make possible to describe both the rigid body behavior and the constant strains;
- invariance is the property of the finite element to have the same physical state irrespective of the orientation of the local axes in relation to which this state is formulated.

Polynomials are the main category of approximation functions used by FEM and can be grouped into the following categories:

- simple polynomials that provide an analytical description of the element's behavior and the functionalities to perform a control over the approximations for the whole domain;
- Lagrange polynomials allow the delimitation of the polynomial coefficients relative to the function values in points on a line;
- Hermite polynomials are used to satisfy both function and derivatives in the finite element nodes.

3. CALCULATION STAGES USING FEM

The analysis of the various structures in terms of resistance using FEM is based on two calculation methods:

- forces – where the unknown values are the resulting forces in the nodes of the physical model of the structure, when it is subjected to a certain load;
- displacements – where the unknown values are the movements that originate in the nodes of the physical pattern of the structure, when it is subjected to a certain load.

The most used is the method of displacements due to the advantages offered by matrix calculation.

Taking into account these general considerations, the main stages during a structural analysis are:

- geometrical modeling;
- mesh preparation: the generation of the finite element wireframe, node numbering and calculation of the geometric properties, such as: coordinates, cross-section area, etc.;
- calculation of the FE equation derivatives;
- variational formula or differentials to describe the required solid body behavior;
- calculation of displacements in respect with nodes movements:

$$u = \sum_{i=1}^{n} u_i \cdot N_i , \tag{1}$$

where u_i are the displacements and N_i – interpolation functions;

- calculation of element matrices;
- assembling finite element equations: identifying the conditions of continuity between adjacent finite elements by primary variables – the relationship between local degrees of freedom and global degrees of freedom; identifying the conditions of equilibrium between the secondary variables – the relations between local components and global components;
- introducing the contour conditions and reducing the system of equations which describes the behavior of the solid;
- solving the system of equations describing the behavior of the studied structure;
- post processing and verification or the results.

4. EXPLICIT DYNAMICS

The explicit scheme (Fig. 4) has the advantage that it does not require the calculation of the stiffness matrix. This significantly reduces the calculation time. The algorithm allows the control of the computational errors

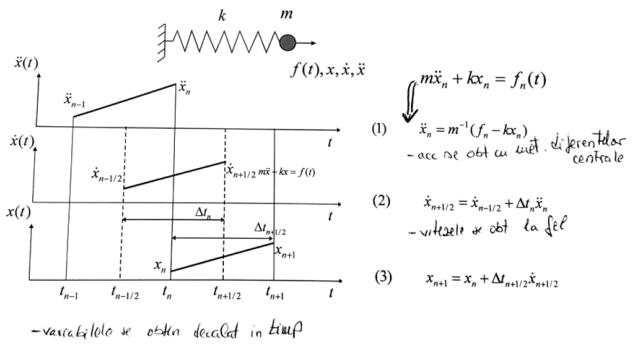

Fig. 4. Explicit integration scheme [6].

and it is successfully used to simulate large impact and deformation problems [6].

The main differences between the implicit and explicit solution schemes are described further.

The implicit solution has the main characteristics:

- does not consider the effect of mass (inertial effects) or damping;
- the analysis is performed using a default solver;
- the solution at each step requires many iterations to assure the equilibrium within a certain tolerance;
- the time steps are generally higher than those used by the explicit scheme;
- it requires a numerical solution to invert the stiffness matrix once or even several times over a time step, which is time consuming for large models.

The explicit scheme is characterized by:

- it takes into account inertial effects and damping;
- it is performed by means of an explicit solver;
- it has no iterations are required because the nodal accelerations are solved directly;
- there is no inherent limit of the time step size;
- the time step must be less than the Courant time step;
- it works relatively easy with contact and nonlinear materials;
- once the accelerations are known at time n, the velocities are calculated at time $n + 1/2$, and the displacements at time $n + 1$. The strains are obtained from displacements and the stress values are obtained using the strain values.

The equations solved when an explicit dynamic analysis is performed are based on the conservation of mass, impulse and energy in Lagrange coordinates. These equations, together with a particular material pattern and the initial set of boundary conditions completely define the computation problem.

General recommendations for model preparation when using the explicit scheme are:

- the size of the elements must be uniform for fine meshed regions;
- time step used is controlled by the minimum size of the finite element;
- models for explicit calculations must have a high quality mesh, as the results propagate across the entire wireframe;
- element size must be controlled by the user on the whole model;
- the element size does not automatically depend on the geometry;

- for explicit analyzes, the location of regions with great strain gradient is constantly changing. The stress waves propagate through the mesh.
- efficiency of the method is improved when refining the mesh;
- transitions between a coarse and a fine mesh must be smooth for maximum accuracy;
- hexahedral mesh has always to be preferred.

5. PAM CRASH SOLVER

PAM CRASH is a finite element solver used for the study of linear and nonlinear materials with large displacements. It has been developed for the automotive industry where it is widely used today. The types of materials studied by the solver are: metals, foams, rubber, composite materials and plastics. The explicit dynamic analysis takes place in increments that presume a very short time. It does not require iterative calculations and matrix inversion and it is widely used to solve quasi-static problems.

The native file of the solver has a pc extension and contains the model definition and a header where the running time, the desired results are defined [6]. The output file has a variable size, depending on the input settings and the required results.

To run a PAM CRASH simulation, the following files are used:

- *model.pc* that includes the geometry and defined boundary conditions;
- *material database.inc* comprising material definition;
- *input.pc* for parameters calculation and files used to launch the solver.

In this study the PAM CRASH files were adapted in order to achieve the kinematic calculation of the robotic structure.

6. COMPUTATIONAL MODEL

For modeling purposes, ANSA pre-preprocessing system was chosen, through which the CAD model of the ABB IRB 760 robot was transformed into a finite element model and the boundary conditions were configured (Fig. 5).

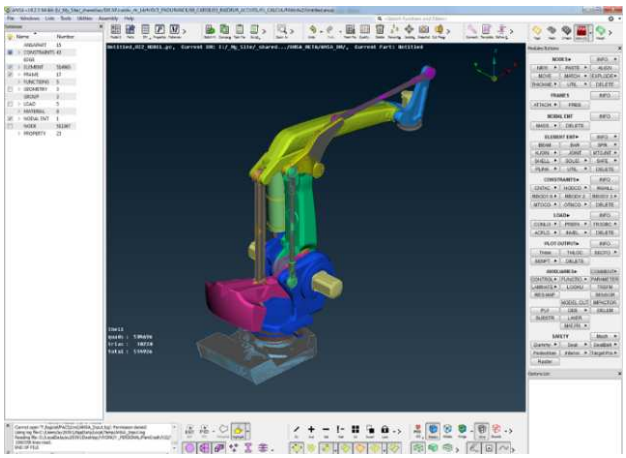

Fig. 5. Preprocessing stages with ANSA.

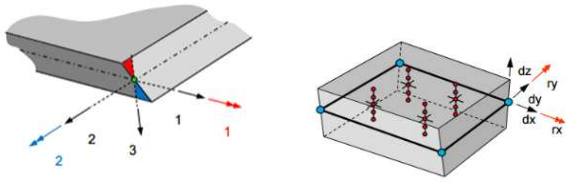

Fig. 6. SHELL 16 elements [6].

6.1. Kinematic and finite element model

The geometry was divided into 514926 finite elements. The element types were chosen from the solver's library (Fig. 6). The size of the elements varied within a minimum/maximum range of 4–12 mm. Several elements connected by common nodes form a part. The properties of the parts were defined in the PID cards.

Shell elements are of several shapes. The most used in this model were SHELL 16.

LS DYNA SHELL 16 is a fully integrated finite element that uses strain interpolation to mitigate the jamming and intensification of the bending feature. It is based on the Reissner-Mindlin kinematic hypotheses: 5 degrees of freedom in the local coordinate system and 6 degrees of freedom in the global coordinate system, 2×2 integration points in the element plane and shear corrections that eliminate numeric errors that may occur [6]. This element is available for both implicit / explicit simulations and uses only quads.

Linking elements were: fixed joints, rotational joints, springs, translational joints and contact elements. The function CONSTRAINED_ND_R_BD defined a rigid link between two points and CONSTRAINED JOINT REVOLUTE/SPHERICAL/TRANSLATIONAL described joints between different components to allow the robot movements. The contact elements simulated relative displacements in joints.

The model also comprises a standard numbering of the assemblies and subassemblies (Fig. 7), thickness property values and also the definition of the oligo cyclic materials for all the parts from which the structure is formed for fatigue calculations.

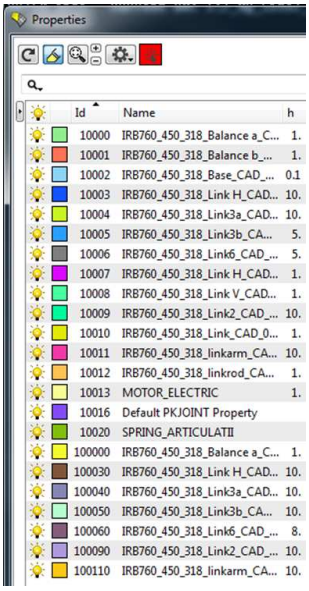

Fig. 7. Model tree in ANSA.

The electric drives and the robot base were connected with the other robot structural elements by means of fixed joints.

6.2. Definition of the kinetic conditions

According to the technical file the robot has four numerically controlled axes (Fig. 8), but neglecting the effector, only three rotational joints with motion laws remained active in the model. The other couplings and free rotation couplers have not been described by initial conditions, limiting their role to keep the assembly connected during operation or having balancing reasons. Although they were included in the model as joints, their movements have been indirectly generated by the joints with imposed motion laws.

The robot was programmed to manipulate a mass attached to axis 4. The motion laws (loads) for the axes 1, 2 and 3 are represented on Fig. 10. These graphs were achieved by using the maximum rotational velocity and the maximum displacements of each axis that is numerically controlled.

Movement

Axis movements	Working range	Maximum speed
Axis 1	+165° to -165°	145°/s
Axis 2	+85° to - 40°	110°/s
Axis 3	+120° to - 20°	120°/s
Axis 4*	+300° to - 300°	400°/s

* +150 rev. to - 150 rev. max

Fig. 8. Axis movements.

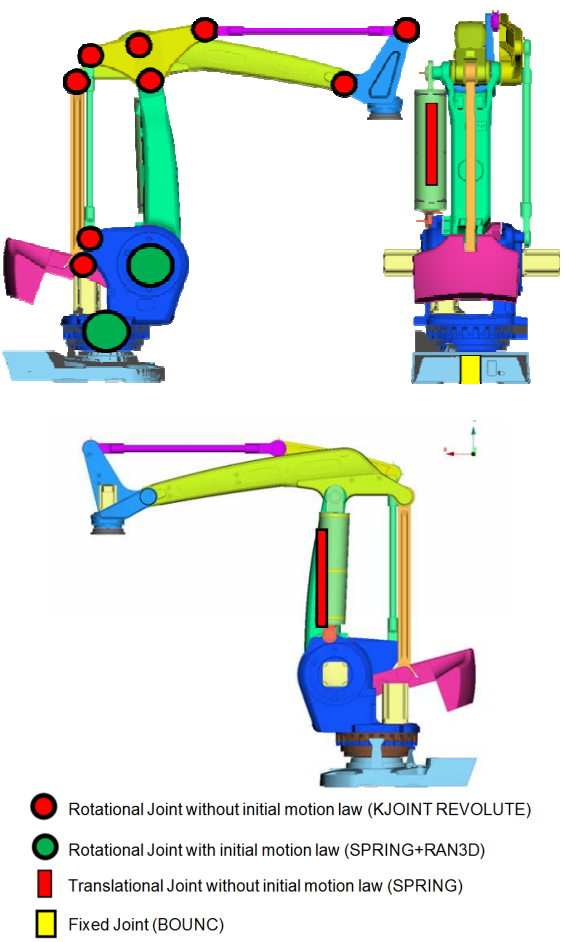

Fig. 9. Robot axes.

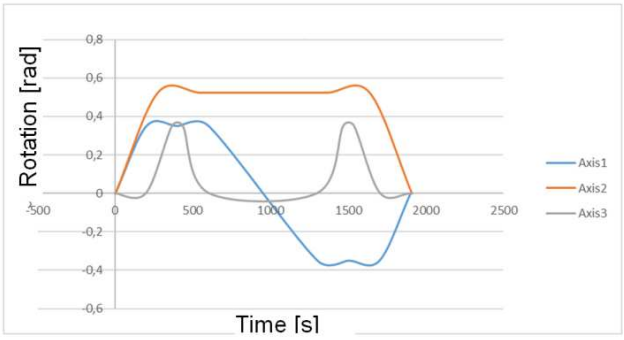

Fig. 10. Motion loads.

6.3. Definition of the rotational joint without initial motion load

In order to define this type of joint (Fig. 11) it is necessary to use:

- rigid bodies on each part of the coupling RBODY [7],
- local coordinate system (Fig. 12) on the RBODY - on the fixed part of the coupling FRAME (Fig. 13) and
- connecting elements between the mobile and fixed parts KJOINT REVOLUTE that allow the rotation around the axis (Fig. 14). Free rotational axes were defined on the local FRAME. All nodes on which link elements are defined are aligned on the rotational axis using the ALIGN command from the MESH menu.

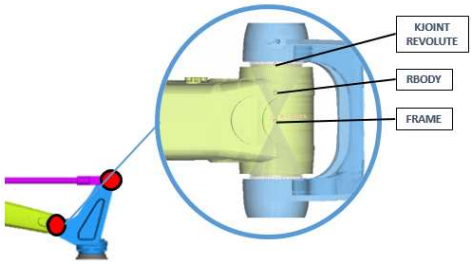

Fig. 11. Rotational joint without motion load.

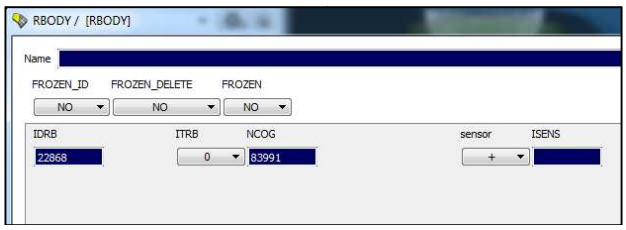

Fig. 12. RBODY definition card.

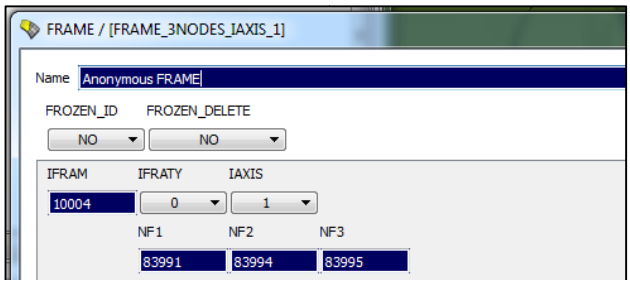

Fig. 13. FRAME definition card.

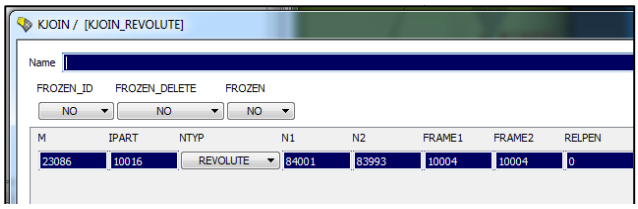

Fig. 14. KJOINT definition card.

6.4. Definition of the rotational joint with initial motion load

To define this type of joint it is necessary to use (Fig. 15):

- rigid bodies on each part of the coupling RBODY,
- a local coordinate system on RBODY - on the fixed part of the coupling FRAME,
- a link element that allows the rotation around an axis by introducing very high stiffness values on the other axes SPRING - this is similar to KJOINT REVOLUTE and it is defined on FRAME LOCAL. Next to the SPRING, a KJOINT REVOLUTE was used in order to stabilize the structure. This was defined on the same FRAME as the SPRING,
- an element that transmits the movement in respect to the motion load (Fig. 16), RAND3D (Fig. 17). All nodes on which link elements are introduced in the model are aligned on the rotational axis using the ALIGN command from the MESH menu.

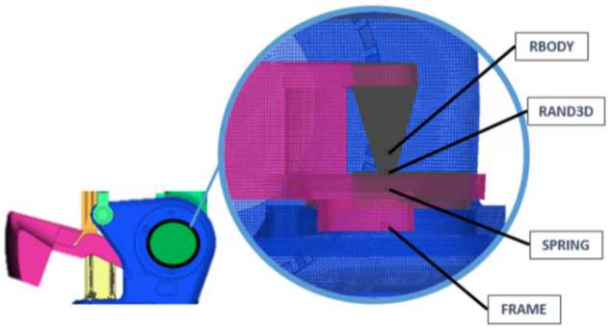

Fig. 15. Rotational joint with motion law.

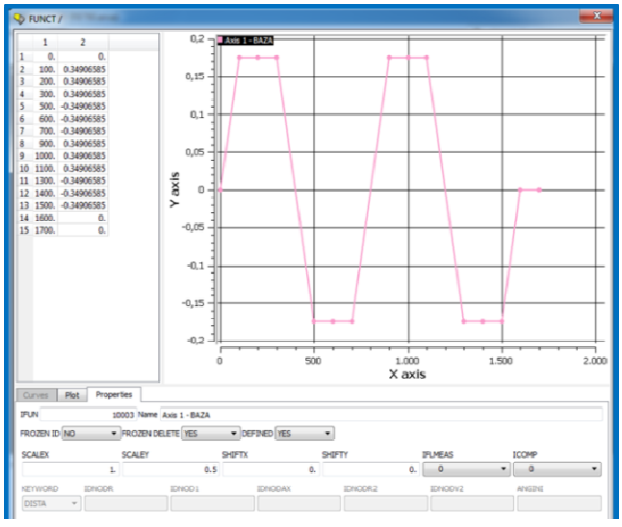

Fig. 16. Motion load for Axis 1. FUNCTION card.

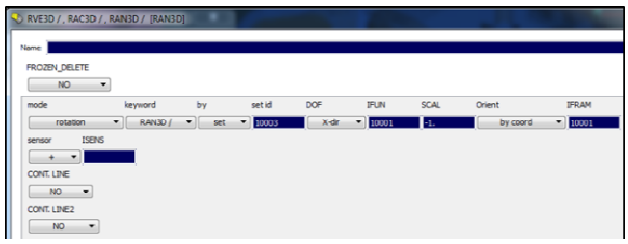

Fig. 17. RAND3D definition card.

Because the duration of the movement directly influences the computational time was limited to 2000 s.

6.5. Definition of the translational joint without initial motion load

To define a translational joint between 2 elements (Fig. 18), following settings were done:

- rigid bodies on each part RBODY,
- a local coordinate system FRAME defined on the fixed part RBODY,
- a SPRING element (Fig. 19) that allows the translation along an axis and defines its stiffness. This property can be changed for any degree of freedom through MATERIAL PART (Fig. 20).

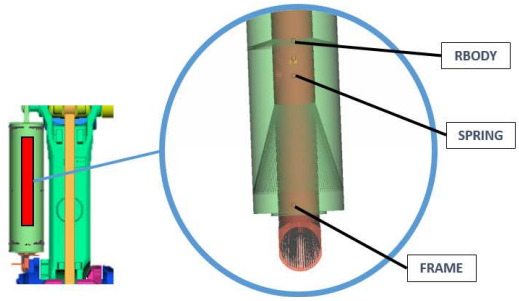

Fig. 18. Translational joint without motion load.

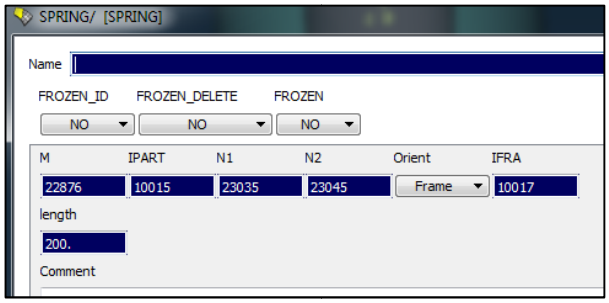

Fig. 19. SPRING card.

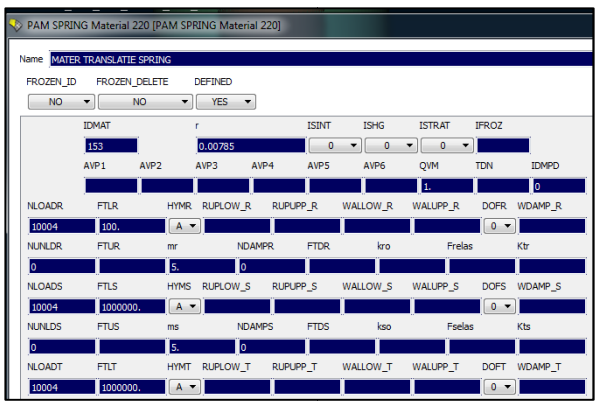

Fig. 20. PAM SPRING MATERIAL definition card.

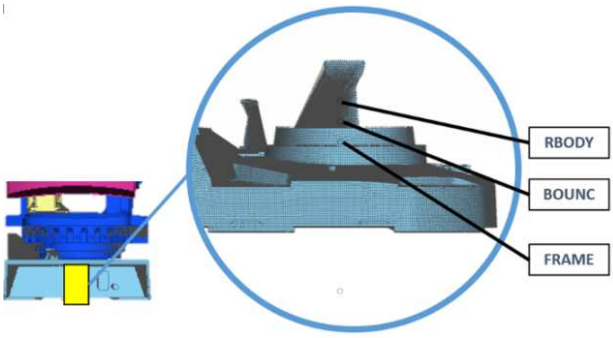

Fig. 21. Body to ground joint.

The kinetic model of the robot also contains fixed joints and couplings, as shown on Fig. 21.

6.6. Definition of the manipulated object

In order to tune the model with the reality the total mass of the robot, as well as the mass of the different structural components were controlled, adding distributed mass to the geometric model so that the model reached the catalog weight. A 69 kg mass of the manipulated object was also attached a concentrated point mass on a node at the top of the robot arm.

7. KINETIC SIMULATION

After all the preparation stages the model was exported to Pam-Crash solver and the launch file was completed with the run-time parameters and the time increments for the result subcases. The server was also specified together with the result types to be saved, as well as the material database used in the simulation.

8. SIMULATION RESULTS

The results chosen when building the header were processed with META-Post processor and mapped on the corresponding geometry (Fig. 23). All the subcases were read, displayed and analyzed (Fig. 22).

As expected, due to the inertial effects and the mass attached at the end of the kinematic chain, the base of the robot recorded a maximum equivalent stress value close to 500 MPa (Fig. 24). This structural component must retain its original form and has to assure reduced values of the maximum displacements in order to maintain the positioning accuracy of the robot, as well as the repeatability during operation.

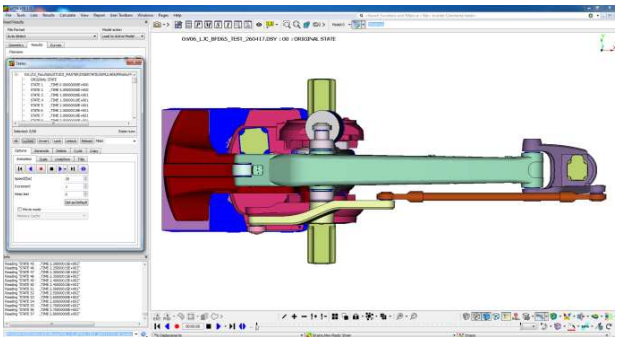

Fig. 22. Robot model loaded in Meta-Post interface.

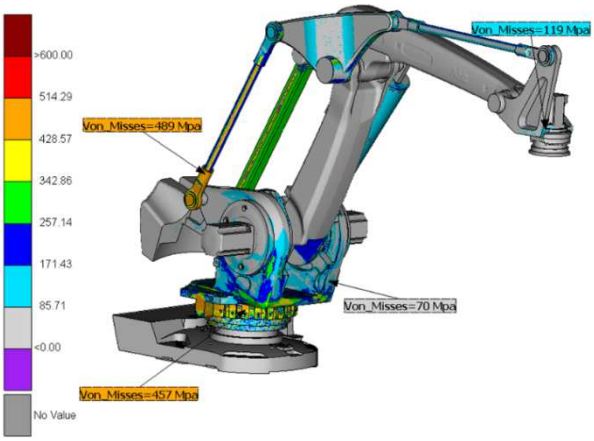

Fig. 23. Equivalent von Misses Stress during operation.

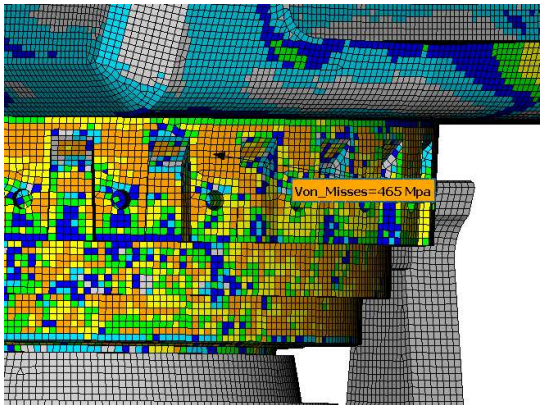

Fig. 24. Detail with the maximum equivalent stress at robot basement.

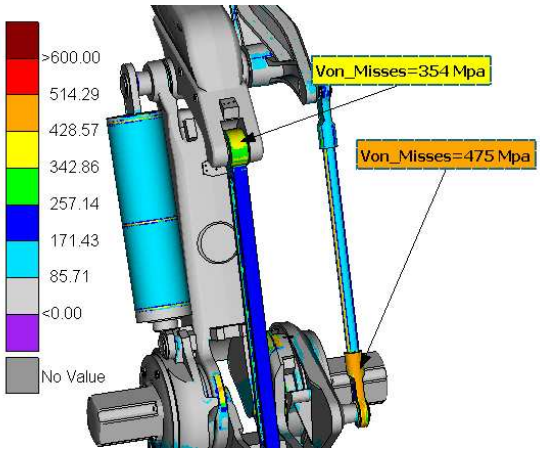

Fig. 25. Maximum equivalent stress on structural elements.

Other areas with high stress values were observed at structural elements that form the closed kinematic chain for balancing reasons (Fig. 25).

The calculation shows that the structure is stiff enough to sustain dynamic forces acting during the robot's operation. However, areas that are fatigue-sensitive, especially those on the closed kinematic chain (bars, triads), were identified and emphasized. This is due to the relatively small cross-section of these parts of the model in comparrison to the rest of the structure, but also due to the draft modeling approach.

9. CONCLUSIONS

The paper presented a case study and described modeling strategies for a transient simulation of a robot model inspired on the industrial ABB IRB 760 one. The analysis prooved that CAE integrated software used mainly in the automotive industry is also a valuable design and research tool for industrial robots by adapting and customizing the simulation methodology.

It is worth to mention that in the present paper the thicknesses and materials used were different from those of the manufacturer; hence a physical test of the real robot may not fully correlate with the results provided by these calculations. The main target of this attempt was to primarly test a set of software applications in a combination used in the automotive industry to study the endurance of an industrial robot structure: ANSA-Pam-Crash-METAPost.

Dynamic analysis has been performed using an explicit solver, which involves time integration and does not require an iterative computation. Due to the relatively large duty cycle the simulation lasted 12 hours on a workstation with 64 GB RAM, 16 Clusters. The result files have about 3.65 GB. The robot has been programmed to manipulate a mass of 69 kg over a distance of 1.5 m at a rotational speed of 110°/s. A comparison was also done between the robot's behavior with and without the manipulated object for the same cycling.

The novelty of this research can be summarized as follows:

- the development of rotational / translational joint models with and without moving law in PamCrash,
- the customization of a dynamic calculation methodology used exclusively in and for the automotive industry on a articulated arm robot and the achievement of reasonable results for its behavior for the actual geometry of the model.

Future work will focus both on the kinematic modeling of the robot to take into account the joint stiffness, and clearance, but also to tune the solver capabilities with the response information in order to drop the computational time.

Acknowledgements: The authors would like to thank BETA CAE Systems SA Greece, to the customer support team for the modeling recommendations as well as for the software license provided for ANSA and META-Post systems.

REFERENCES

[1] Z. Fan, X.Yu, Q. Zhang, S. He, G. Chen, J. Du, Y. Wen, *Fatigue Life Estimation for Simply-supported Pipeline of Robots under Hybrid Excitation*, International Journal of Fatigue (2017), accessed 2018.03.05, doi: `https://doi.org/10.1016/j.ijfatigue.2017.11.002`.

[2] Shuxiang Guo, Yanlin He, Liwei Shi, Shaowu Pan, Kun Tang, Rui Xiao, Ping Guo1, *Modal and fatigue analysis of critical components of an amphibious spherical robot*, Springer Microsyst Technol, Published online 29 July 2016, doi: 10.1007/s00542-016-3083-0.

[3] Y. Lin, G. Shuai, Xi Fengfeng *Preferable Workspace for Fatigue Life Improvement of Flexible-Joint Robots Under Percussive Riveting*, ASME Journal of Dynamic Systems, Measurement, and Control, April 2017, Vol. 139, pp. 041012-1, 041012-7.

[4] F.P. Beer, Jr. E. Russel Johnston, DeWolf J.T., Mazurek D.F., *Mechanics of Materials*, Seventh Edition, Mc Graw Hill, USA, 2015.

[5] V. Oleksik, *Analiza dinamică a structurilor* (Dynamic analysis of structures), published by CCIMN Sibiu, 2009.

[6] ESI Group – *Pam Crash User Guide Documentation*, 100-102 Avenue de Suffren, Paris, Frace, 2017.

[7] BETA CAE System *Ansa 17 Pre-processor User Guide*, Kato Scholari, Greece, 2017.