

Alzbeta Sapietova – Milan Saga – Pavol Novak *

MULTI-SOFTWARE PLATFORM FOR SOLVING OF MULTIBODY SYSTEMS SYNTHESIS

The paper deals with design and application of multi-software platform for solving of mechanical multi-body system problems. The Adams working interface and open architecture of Matlab programming language enable share common data during parallel run of simulations. This process was used for design and implementation of evaluation algorithm and optimization of technical equipment parameters in terms of their mechanical properties. The goal is to present the mapping process of the working space of the chosen manipulating equipment and application of the optimizing approach to find its geometry parameters. Multi-software technique will be applied for computational realization i.e. Adams and Matlab.

Key words: coupled mechanical system, optimization, Matlab, ADAMS

1. Introduction

Modeling can play an important role in the analysis and design of flexible multibody systems. Recent developments in commercial FEA packages allow to the designer to make the detailed analysis of composites structures, too [1, 2].

The problem during the production machine synthesis solving is to find the suitable construction and the corresponding suitable parameters of the mechanism (i.e. materials, dimensions of body shapes and cross-sections, algorithms for the movement controlling, etc.). The parameters must fulfill some functional and technological requirements. These requirements may relate only to the position of the mechanism elements. In this case we call it the geometrical synthesis. In the other cases we must deal with the velocity and acceleration requirements. The methods solving these problems are the scope of the kinematics synthesis.

The dynamical requirements (i.e. balancing, reaction forces reduction, motion irregularity reduction, required motion properties, etc.) are solved by the dynamical synthesis [3].

The area of the mechanism synthesis has been primarily defined as a relationship between the input and the output (1:1). However, this area is more complicated for the production machine mechanisms and it covers the whole mechanics and the control area. The solutions can be characterized as multiple inputs – outputs (M:N). Therefore it is needed to extend the traditional mechanic techniques with the techniques focused on the general design theory [3].

2. Files preparation for the kinematics synthesis of the rigid system virtual prototype

The solution objective is to design and implement the algorithm of the evaluation of the operating gear parameters from the mechanical properties view [4]. The approach is based on the solution of the fixed solid system (FSS) virtual prototypes by means of the computer simulation, using the kinematics and the optimization [5, 6]. There is tendency to eliminate the time-consuming analyses so that the program interconnection of the ADAMS and MATLAB systems will be designed and implemented. For the inverse kinematics problem solving the virtual prototype (VP) manipulator robot was chosen (Fig. 1). The solution contains the following processes:

- Virtual prototype preparation in the ADAMS/View environment,
- modification of the simulation,
- creation of an “*.acf” file,
- data export into an “*.adm” file,
- creation of programs in MATLAB environment,
- inverse kinematics problem,
 - the creation of the point of interest working positions map,
 - the movement trajectory specification for the point of the interest,
 - the selection of the points from the working positions map that copy the prescribed trajectory in the best way,
 - the optimization of the rigid system element lengths.

* Alzbeta Sapietova, Milan Saga, Pavol Novak

Department of Applied Mechanics, Faculty of Mechanical Engineering, University of Zilina, Slovakia, E-mail: alzbeta.sapietova@fstroj.uniza.sk

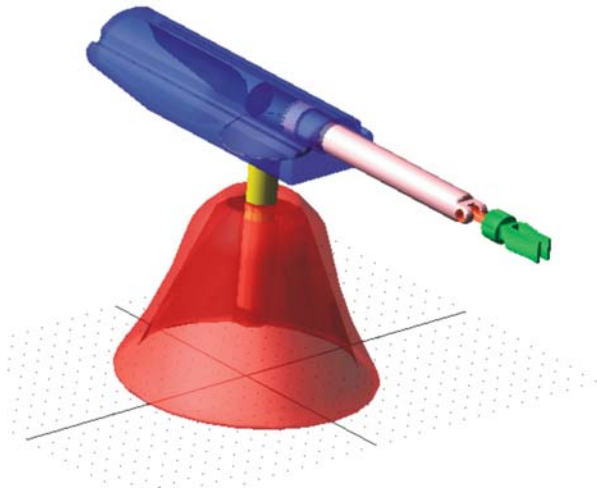


Fig. 1 Virtual prototype manipulator robot in environment ADAMS/View

2.1 Virtual prototype preparation in the ADAMS/View environment

The particular elements of the manipulator robot were designed in the ADAMS/View environment. Kinematic schema of the manipulating equipment has four movement degrees of freedom (Fig. 2). Quantity “ v ” is speed of motion and quantity “ ω ” is angular velocity.

During the creation of virtual prototypes the models in the ADAMS environment are saved in the various file formats according to their purpose. The manipulator robot was created in the binary file format (model.bin). This database contains the information about the working space configuration, all properties of one or more models and the analyses and the simulations results too.

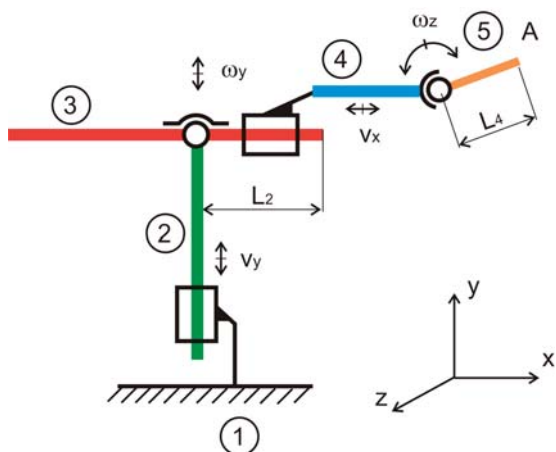


Fig. 2 Kinematic schema of manipulating equipment

The binary format allows the quick writing and loading of the data and it can be transferred between computers with the different operating systems but it does not allow their reading and editing.

2.2 Modification of the simulation

In the modification of the simulation, the prescribed process of the movements for all working cycles is defined. Modification of the simulation is created so that motion was directed gradually. In the particular joints the following movements are prescribed (Fig. 2 , Fig. 3):

MOTION_1 [“-1*time”]	% displacement PART_2 - v_y
MOTION_2 [-200]	% persistence of the PART_2 in the given position
MOTION_3 [“1d *(time-200)”]	% rotation of the PART_3 - ω_y
MOTION_4 [180d]	% persistence of the PART_3 in the given position
MOTION_5 [“1*(time-380)”]	% displacement PART_4 - v_x
MOTION_6 [100]	% persistence of the PART_4 in the given position
MOTION_7 [“-1d*(time-480.0)”]	% rotation of the PART_4 - ω_z

When the solid moves and it needs to stop afterwards, the zero movement must be activated for it in the time of the stop. Therefore more movements (MOTION_1, MOTION_2) are included into the JOINT (Fig. 3).

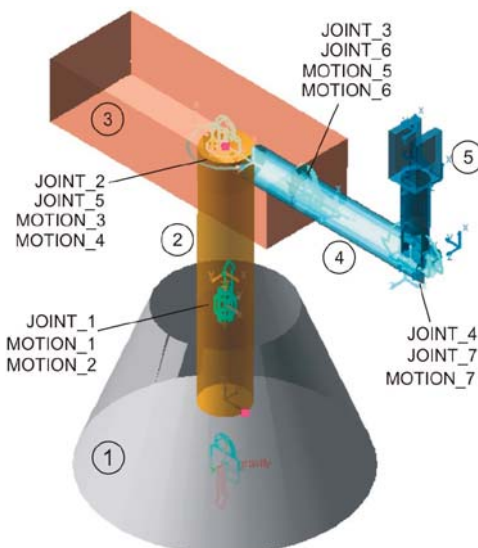


Fig. 3 Joints and motions of the manipulating equipment

Functionality of the virtual prototype (VP) is verified by the simulation in ADAMS/View environment and the kinematic parameters are calculated. Diagrams in Figs 4 to 6 illustrate trajectory of centre of gravity of particular parts.

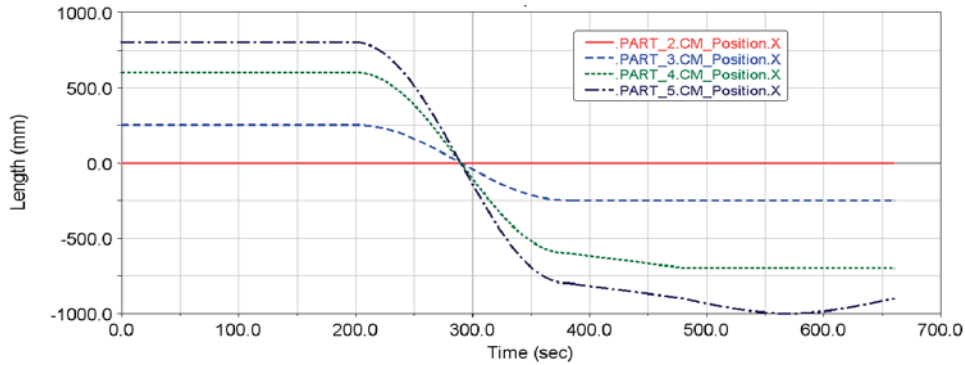


Fig. 4 Trajectory of centre of gravity of particular parts - position x

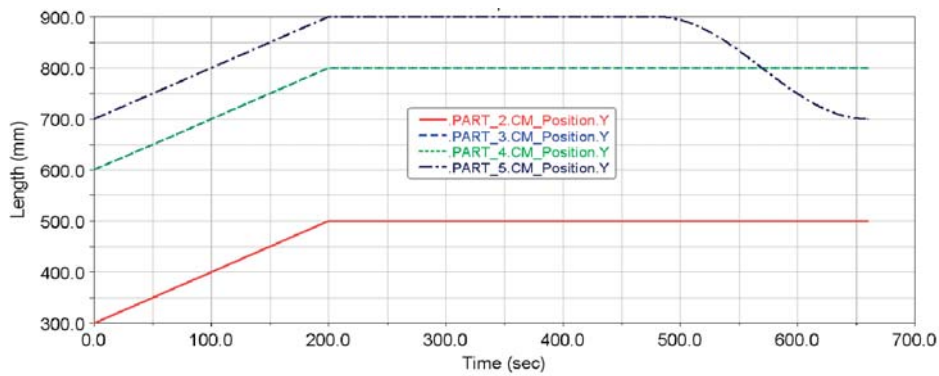


Fig. 5 Trajectory of centre of gravity of particular parts - position y

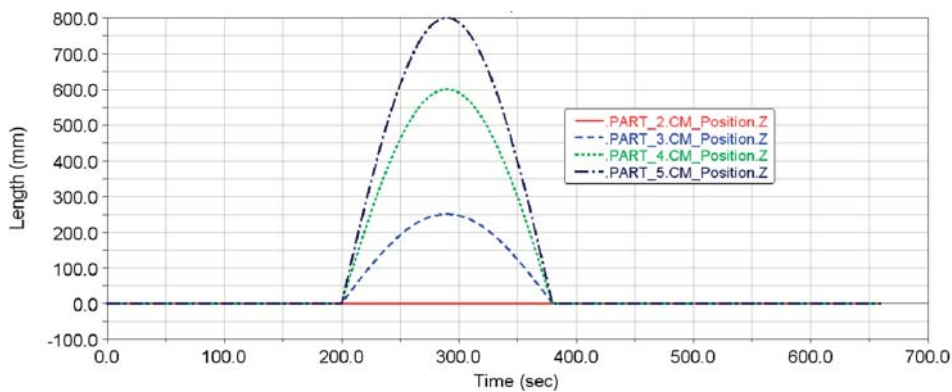


Fig. 6 Trajectory of centre of gravity of particular parts - position z

2.3 Creation of an “*.acf” file

Values for simulation modification instruction are stored into the new created file with “*.acf” suffix. This file can be imported into ADAMS environment with the aim to run ADAMS/Solver with the prescribed movement simulation modification and the consequent system solution.

2.4 Data export into an “*.adm” file

Model in the “*.adm” format is the record in its most saving form for sending to the ADAMS/Solver environment. It contains the data only for one model; it does not store the environment configuration. The advantage is the ASCII format in which the “*.adm” file can be comfortably read, edited and transferred between com-

puters with the different operating systems. This file can be again imported into ADAMS environment in order to run the ADAMS/ Solver.

2.5 Creation of programs in MATLAB environment

Programs created in the MATLAB environment control the computation approach. These programs are stored in the following files:

- *Optim_GENERAL.m* - this file contains optimization methods with the call of *dona.m*,
- *dona.m* - the file for computation of the objective function and for the solution of the point positions with the minimal distance from the prescribed trajectory. This program calls the subprogram *runAdams.m*,
- *runAdams.m* - it runs the ADAMS system and creates the result file *model.res*,
- *opt_uhl.m* - it optimizes the output parameters, e.g. angles of rotation in order to obtain the continuous movement.

2.6 Inverse kinematics problem

The inverse kinematics problem solves the suitable input kinematic parameters multibody of the systems (MBS) for obtaining the described movement [4]. The graphical diagram of the particular program relations is shown in Fig. 7. Assuming the stepping motor realistic usage with step of 5 degrees, all working positions of the manipulator robot endpoint "A" are mapped (Fig. 8). Prescribed trajectory of the point "A" is a helix (Fig. 9). From the obtained working position map the coordinates of those points were chosen which correspond with the prescribed trajectory in the best way and which meet the condition of the continuous movement MBS at the same time.

The input kinematics parameters ensuring the given element prescribed movement MBS are the solution results. The chosen results of the rotation angle values and displacements are shown

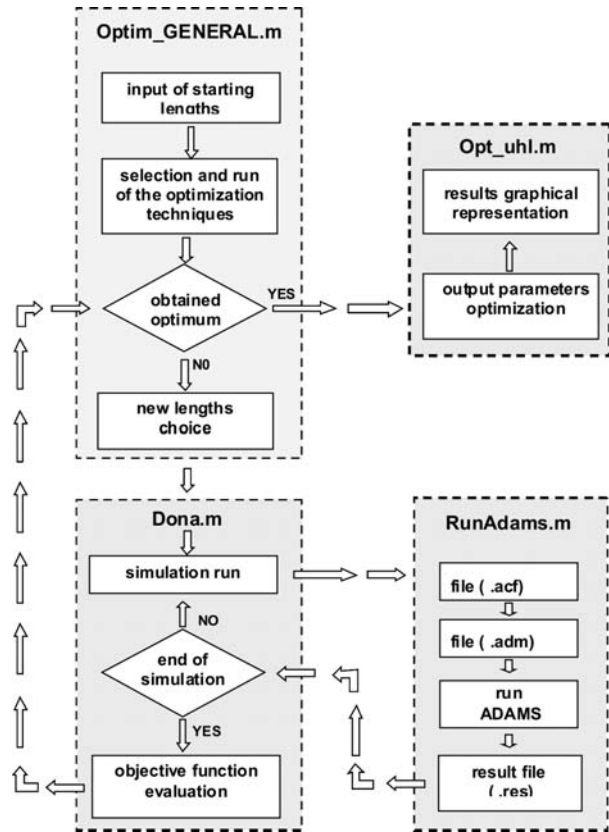


Fig. 7 Graphical representation of the inverted kinematics problem solution

in Tab. 1. Their number depends on the solution step size. Parameter "r" is minimal distance of the solved point from trajectory.

The genetic algorithm method was used. The objective function value was a numerical zero. In order to obtain the continuous movement the optimal length value was used in the program *opt_uhl.m*.

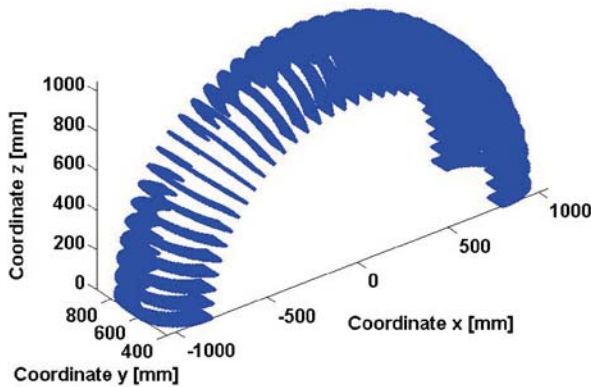


Fig. 8 Point "A" - map of working positions

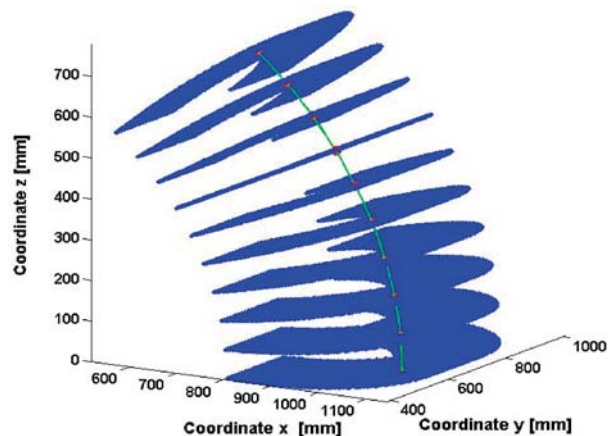


Fig. 9 Point "A" - map of working positions

2.6.1 Optimization of geometric parameters of a manipulator.

From the working positions map the points closest to prescribed trajectory were selected. Due to closer following of prescribed trajectory by end point of the manipulator, optimization of lengths of selected parts was involved. The solution algorithm was as follows:

- working positions map for point “A” contains n points, i.e. positions of point “A” on real trajectory k_1 ,
- on the prescribed trajectory k_2 n key points were defined - required positions of point “A”.
- Next, we minimize objective function F , which is a vicinity criterion of trajectories k_1 and k_2 :

$$F(L_2, L_4) = \sum_{i=1}^n \sqrt{dx_i^2 + dy_i^2 + dz_i^2} \rightarrow \min ,$$

where

$$dx_i = [x_i(L_2, L_4)]_{k_1} - [x_i]_{k_2},$$

$$dy_i = [y_i(L_2, L_4)]_{k_1} - [y_i]_{k_2},$$

$$dz_i = [z_i(L_2, L_4)]_{k_1} - [z_i]_{k_2}.$$

where $[x_i, y_i, z_i]_{k_2}$ are coordinates of point “i” on the prescribed trajectory and $[x_i(L_2, L_4), y_i(L_2, L_4), z_i(L_2, L_4)]_{k_1}$ are coordinates of point “i” on the real trajectory for given design iteration. Lengths L_2 and L_4 are design variables (Fig. 2). Initial values of design variables were $L_2 = 500$ mm, $L_4 = 200$ mm. The optimal values of the manipulator robot element lengths are: $L_2 = 515.92$ mm, $L_4 = 191.85$ mm (Fig. 2).

Pattern search function from Matlab Optimization Toolbox was used to design an optimization problem. The algorithm of this

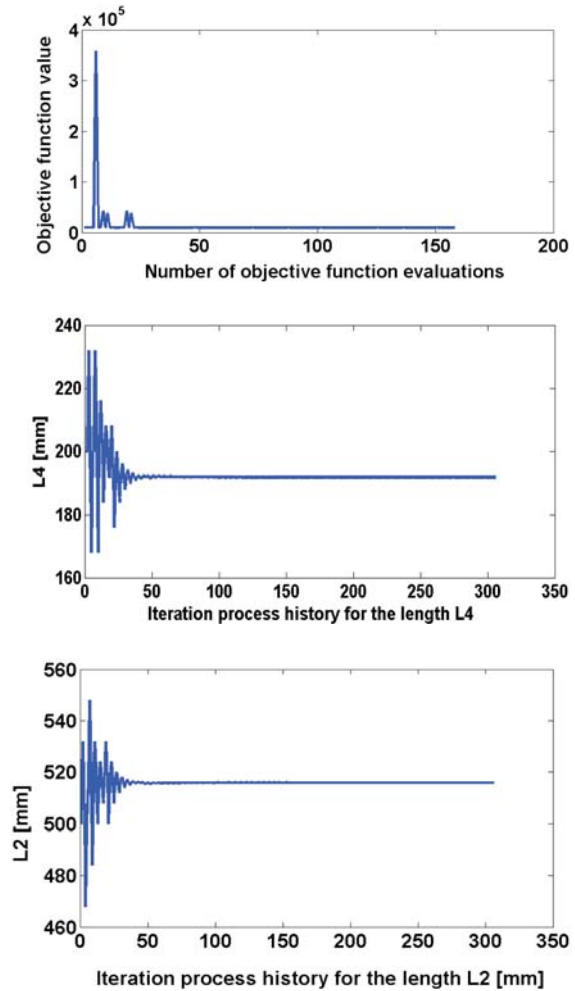


Fig. 10 Iteration process history

Evaluation of monitored values in the MBS synthesis.

Table 1

Dispacement Part_2 (mm)	Angle Part_3 (°)	Dispacement Part_4 (mm)	Angle Part_5 (°)	Position x (mm)	Position y (mm)	Position z (mm)	Min distance (mm)
60	2.5	50	80	1022.09	549.95	44.62	0.07
80	7.5	30	80	1014.31	569.95	133.53	17.85
100	10	30	80	1007.52	589.95	177.65	8.92
10	15	15	45	987.91	610.0	264.71	8.92
30	17.5	15	45	975.42	630.0	307.55	17.85
50	22.5	15	45	944.91	650.0	391.39	0.23
70	27.5	15	45	907.20	670.0	472.25	17.85
90	30	15	45	885.73	690.0	511.38	8.92
0	35	35	10	838.04	710.04	586.80	8.92
20	37.5	35	10	811.65	730.04	622.80	17.85
40	42.5	35	10	754.28	750.04	691.17	0.07

function is based on GA principles. Iteration history and solution history for design variables are shown in Fig. 10.

3. Conclusion

The goal of the paper was to suggest and apply the approach for the synthesis solution and the consequent manipulation device analysis using the interconnection of ADAMS and MATLAB programs. By reason of significant requirements for accuracy, economy, reliability and operating lifetime of technical equipment it is necessary to consider several parameters that secure the optimal product.

For that reason the next step is interconnection between simulation models and software for numerical optimization that enabled iterative improvement of the designed system. This was made on the base of exact optimization methods. From the experiences obtained during the problem solution, the next goals can be formulated; to modify the program in order to shorten the computation times and to use the whole range of possibilities offered by the ADAMS software.

Acknowledgement

The part of the results of this work has been supported by VEGA grant No. 1/1259/12 and the VEGA grant No. 1/1000/12.

References

- [1] KORMANIKOVA, E., RIECKY, D., ZMINDAK, M.: Strength of Composites with Short Fibers. In: Eds, J. Murin: Computational Modelling and Advanced Simulations, Springer Series: *Computational Methods in Applied Sciences*, vol. 24, 2011, ISBN 978-94-007-0316-2
- [2] ZMINDAK, M., RIECKY, D.: SOUKUP, J.: Failure of Composites with Short Fibers. *Communications - Scientific Letters of the University of Zilina*, No. 4, 2010. ISSN-13354205
- [3] VALASEK, M.: *Synthesis of Mechanisms of Production Machines*. In *22th Conference with Intern. Participation, Computational Mechanics*, 2006, Plzen : TYPOS-Digital Print, s.r.o., ISBN 80-7043-477-5, p. 29-38.
- [4] ARNOLD, M., SCHIEHLEN, W.: Simulation Techniques for Applied Dynamics, *CISM Courses and Lectures*, vol. 507, Italy : Springer Wien : New York, 2008, p. 313, ISBN 978-3-211-89547-4.
- [5] DE JALLON, J. G., BAYO, E.: *Kinematic and Dynamic Simulation of Multibody Systems. The Real-Time Challenge*, Springer-Verlag, New-York, 1994, p. 440, ISBN 0-387-94096-0.
- [6] SEGLA, S., CIUPITU, L., REICH, S.: Optimization of a Spring Balancing Mechanism for Parallelogram Robot Mechanisms. In *J. Mechanisms and Manipulators*, ISSN 1583-4743, 2006, vol. 5, No. 2, 2006, pp. 43-48.