Proceedings of the ASME 2010 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference
IDETC/CIE 2010
August 15-18, 2010, Montreal, Quebec, Canada

DETC2010-28994

ROLE OF AUTOMATED SYMBOLIC GENERATION OF EQUATIONS OF MOTION IN MECHANISM AND ROBOTICS EDUCATION

Hrishi Shah, Sumit Tripathi, Leng-Feng Lee
Graduate Students
Mechanical and Aerospace Engineering
State University of New York at Buffalo
318 Jarvis Hall, Buffalo NY 14260
E-mail: hlshah, st69, llee3@buffalo.edu
Web: http://www.eng.buffalo.edu/~hlshah, st69, llee3

Venkat Krovi
Associate Professor
Mechanical and Aerospace Engineering
State University of New York at Buffalo
318 Jarvis Hall, Buffalo NY 14260
E-mail: vkrovi@buffalo.edu
Web: http://www.eng.buffalo.edu/~vkrovi

ABSTRACT

In recent years there has been a significant increase in the variety and complexity of Articulated-Multi-Body-Systems (AMBS) used in various applications. There is also increased interest in the model-based design-refinement and controller-development, which is critically dependent upon availability of underlying plant-models. Kinematic and dynamic plant-models for AMBSs can be formulated by systematic application of physics postulates. This process, in its various variants, forms the basis of various mechanisms/robotics courses. However, the type and complexity of the example systems is often limited by the tractability of first generating and subsequently analyzing complex equations-of-motion. Nevertheless, using simpler examples alone may sometimes fail to capture important physical phenomena (e.g. gyroscopic, coriolis).

Hence, we examine the use of some contemporary symbolic- and numeric-computation tools to assist with the automated symbolic equation generation and subsequent analysis. We examine a host of examples beginning with simple pendulum, double pendulum; building up to intermediate examples like the four-bar mechanism and finally examine the implementation of 3-PRR and 3-RRR planar parallel platform mechanisms. The principal underlying philosophy of our effort is to establish linkage between traditional modeling approaches and use of these contemporary tools. We also try to make a case for use of automatic symbolic computation and manipulation as a means for enhancing understanding of both basic and advanced AMBS concepts. Lastly, we document our efforts towards creation of self-paced tutorials and case-studies that serve to showcase the benefits.

INTRODUCTION

Over the past few decades, several seminal textbooks [1-3] have addressed the mathematical modeling and analysis of kinematics and dynamics of articulated multibody systems (AMBS). In their simplest form, the governing EOM take the form of a system of Ordinary Differential Equations (ODEs). However, there are many factors that can quickly introduce complexity in these governing equations. First, and perhaps the greatest, source of complexity comes from the effects of finite rotations in two- and three-dimensions, which introduces trigonometric complexity (in the form of sine and cosine terms). This modeling complexity is amplified in the transition from linear (1D) systems to planar (2D) systems and ultimately to spatial (3D) systems. Second, most real-life multibody systems possess one or more closed kinematic loops, typically to enhance their stiffness and payload capacity. Such closed-kinematic loops can help reduce overall actuation requirements by creating constraints within system degrees-of-freedom. However, these algebraic constraints interact with the underlying systems of ODEs of the unconstrained systems to create systems of Differential Algebraic Equations (DAEs). Hence, both the initial modeling as well as subsequent performance analysis tend to be difficult in such multibody systems.

Nevertheless, effectively modeling and analyzing the kinematics and dynamics of all such systems within a computer-based model is critical for the apriori prediction of the overall system response. From a design perspective, accurate and computationally-efficient simulation models are vital for rapid design-refinement through iterative simulation-based parametric studies. From the control perspective, the same model can also help implement more effective model-based nonlinear control strategies. Thus, the fundamental challenge in such systems remains: "Given a description of a mechanical system in terms of the relative physical layout, interconnections, and mechanical properties, how we can formulate the kinematic or dynamic Equations-Of-Motion (EOM) and characterize the system response?"

The variety of formulations that exist for multibody systems can be daunting. Such variety arises from the interplay between (i) the multitude of problem tasks that can be addressed, (ii) the varying levels of analysis, and the numerous possible system configuration descriptions. The designer may seek to address forward or inverse problems for such systems, operating in the kinematic or dynamic regimes, with system configurations modeled in terms of a variety of coordinates (absolute, relative, mixed). Oftentimes, selection of specific coordinate descriptions for systems offers unique advantages and disadvantages for specialized problem tasks/analyses. For example, in systems with joint-based actuation, a relative joint-coordinate-based formulation simplifies the determination of the (external) actuation forces for inverse-dynamics problems. However, additionally determining the internal pin-reaction forces is easier in some form of extended coordinate system (for example, absolute Cartesian coordinates of each link) with suitable constraints within an augmented Lagrange formulation.

Traditionally, the ability to select and switch the formulation, depending on the task at hand has created challenges – oftentimes requiring a reformulation of the EOM from scratch. It is also worth noting that practical limits on system-size are often encountered when using certain existing formulations (such as the Lagrangian formulation) to derive EOM of increasingly-complex systems. For all but the simplest problems, however, this task can be laborious and error-prone. The complexity encountered in real-life multibody systems can very easily take many man-months of effort to develop and validate by hand.

In our work, we explore the use of MapleSim [4], a Maple toolbox to facilitate the rapid and automated creation of the symbolic EOMs of large-scale articulated multibody models. We will exploit for ability to create kinematic and dynamic EOMs, using the user's preferred formulation and coordinates within a systematic and automated symbolic implementation.

However, there remains a question of applicability and accuracy of this new found capability which we will systematically examine. We will study this in the context of several case studies, beginning with simple pendulum, double pendulum; building up to intermediate examples like the four-bar mechanisms, and finally examine the implementation of 3-PRR, 3-RRR planar parallel platform mechanisms. These case studies engender many of the complicating factors entering the equations-of-motion, i.e., there are spatial articulated system with active and passive joints and feature multiple closed kinematic loops.

The rest of the paper is organized as follows. In Section 2, we discuss the background as well as challenges in conventional approach and how contemporary tools can help alleviate some of these challenges. In Section 3, we layout the staged implementation of our tutorials, followed by several case studies used in the tutorials. Finally, we discuss some of the issues we faced in implementing the tutorial.

BACKGROUND & CHALLENGES

Traditionally, many of the concepts and ideas for AMBS (including the study of kinematics and dynamics), are delivered in a classroom-based lecture. In this setting, mathematical formulations of the mechanism are usually emphasized and students are required to formulate the equations governing the kinematics and dynamics of some simple mechanisms and then solve these using algebraic techniques. The main advantage of this approach is that it permits the student to understand the fundamental theory underlying the analysis as well as get a handle on the formulation that forms the basis for the analysis of more complicated mechanisms. Thus, with a grasp of the basic concepts and formulations, students can implement the techniques algorithmically by suitably programming [5-6]. However, the complexity of the analyzed mechanism imposes limitations for the analytical method. For example, the formulation of a set of equations for a simple four bar mechanism is manageable for links with center of mass along the link lengths. However, if the shape of the linkage is complex or the number of links increases, the formulation become more complicated and time consuming. Thus, the analytical method is most often typically limited to simple two-dimensional mechanisms and links with relatively simple geometries.

Many of these problems can be alleviated by using computer aided analysis software tools (e.g. ADAMS, VisualNastran, etc.) which are available to support the simulation based study of AMBS. The main benefits of this method are that the students can analyze more complex mechanisms with detailed link geometries, obtain quick results and compare many possibilities prior to selecting best mechanism by permitting the detailed visualization of virtual mechanisms, giving the student a better understanding of the motion of the mechanism, the path of a specific point, and the functionality of the mechanism. The principal disadvantage is that the formulations of the kinematic and dynamic analysis of the mechanism are completely hidden from the student. The black box approach to the underlying governing equations can in many cases hinder understanding of the concepts behind many of the mechanisms.

It is to overcome the limitation of both these approaches that we examined the third automated symbolic modeling approach. Automated symbolic approaches derive their many advantages by eliminating the need for manual EOM computation and subsequent manipulation which was error prone and time consuming. Two trends that have favored the adoption and rapid proliferation of the symbolic computation are: (i) the availability of low-cost PC based symbolic manipulation tools like Maple; and (ii) the capability of integrating multiple functionalities into a unified environment like MapleSim.

However, there are several pedagogic issues that could hinder a direct deployment of technological tools:

A. Currently the use of MapleSim needs "expert users" who can not only model, but also analyze the results for their correctness. While tutorials are made available by the vendors of these tools, they are targeted at a more experienced user (typically with a graduate level knowledge of AMBS). These traditional tutorials may assume a certain level of both mathematical sophistication and engineering experience from the user.

B. Novitiate robotics students may tend to have difficulty understanding both technical (theoretical) concepts as well as their technological implementation. Moreover, it is crucial that student gleans a greater insight into the problem and is better equipped to make engineering judgments from the information obtained from the use of such software.

It is to promote this type of greater understanding that we are creating a series of self-paced MapleSim tutorials with the target audience being the students of the course MAE 413/513: "Robotic Mobility and Manipulation" at the State University of New York at Buffalo. The goal is to reinforce the ideas and concepts originally presented in the course by paralleling the course material with these tutorials. The overall desired outcome includes improving the overall understanding of mechanisms by the students and accelerating their learning experience without increasing the lecture hours.

TUTORIALS IMPLEMENTATION [7]

Traditionally, many of the concepts and ideas behind mechanism theory (including the study of kinematics and dynamics of mechanisms), are delivered with simple examples in a didactic classroom setting.

In MAE413/513 we begin with the formulation of dynamic EOMs of various simple mechanical systems such as simple pendulum using the Newton Euler and Euler Lagrange methods. This is followed by the extraction of dynamic matrices and structural properties. After obtaining the EOM, student then proceed to perform various forward and inverse dynamics simulations in MATLAB using numerical integration routines (ode5 for fixed time-step solver and ode45 for variable time step solver).

While addressing the basic formulations, many assumptions were made in order to simplify the calculations. However, it was unclear as to how situations involving more complex link geometries could be handled. For example, if the given links are not slender (shown in Fig. 1) and/or if the center of mass is not at the geometric centroid of the link, then the problem cannot be directly handled using the methods taught in class.

Many of these problems can potentially be alleviated using Maple with MapleSim toolbox for block-modeling modeling and automated equation generation of various AMBS and its variants. The main benefits of this tool are that the students can analyze more complex mechanisms with detailed link geometries, obtain quick results and compare many possibilities prior to selecting best mechanism by permitting the detailed visualization of virtual mechanisms, giving the student a better understanding of the motion of the mechanism, the path of a specific point, and the functionality of the mechanism. The principal disadvantage of other tools employing automation is that the formulations of the kinematic and dynamic analysis of the mechanism are completely hidden from the student. The black box approach to the underlying governing equations can in many cases hinder understanding of the concepts behind many of the mechanisms. However, MapleSim allows the user to extract the equations in any form desired and export them to other software (e.g. MATLAB), if required.

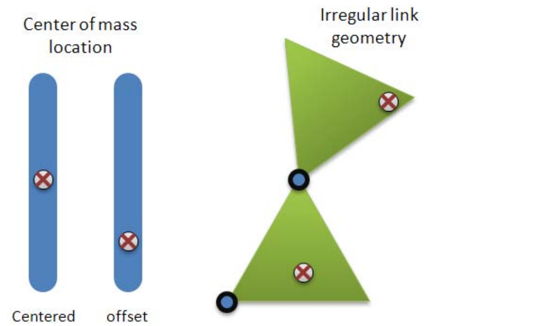


Fig. 1: In formulating equations of motion of a mechanism, simplification/assumption such as inertia of rod and center of mass is at the middle of the rod is often used. Such formulation cannot be directly applicable to irregular shaped linkages.

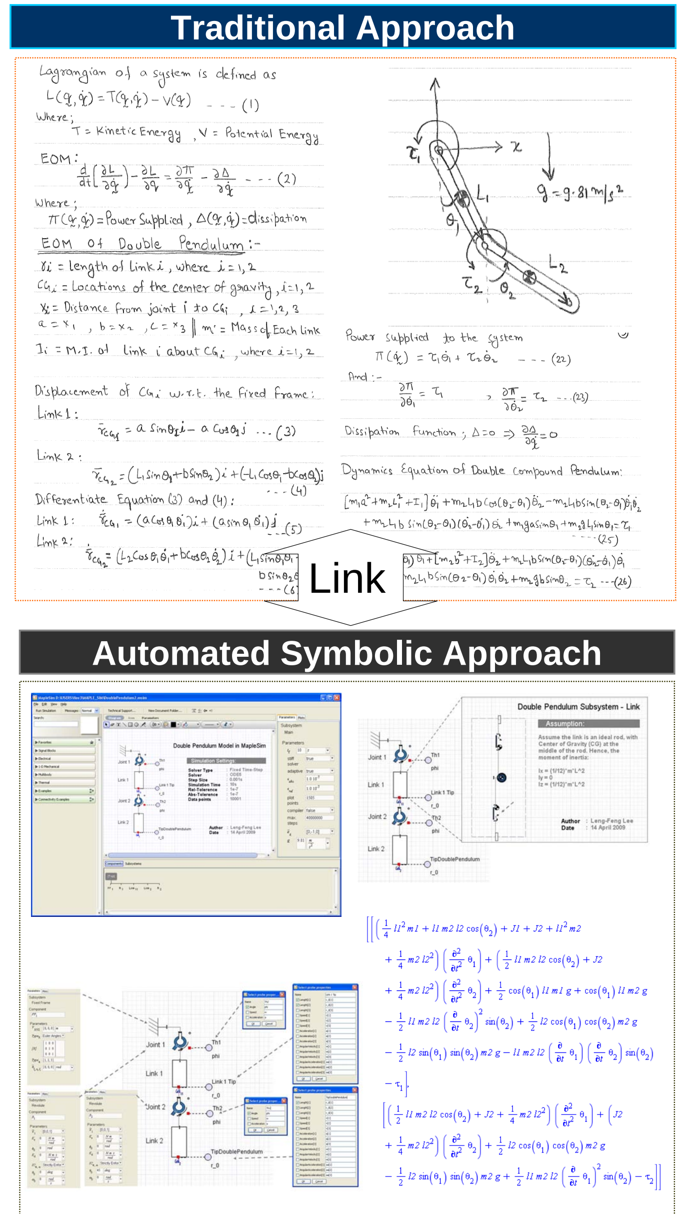


Fig. 2: Traditional approach and automated symbolic approach to obtain equation of motion of a double pendulum. We wish to provide linkage between these two approaches.

Our goal is to create a linkage between the Automated Symbolic Computing approach and the traditional analytical approach so that the student can derive benefits from both – a better understanding of the problem as well as greater proficiency in different methods available to solve it.

The lecture coverage of the course emphasized the use of the traditional low-resolution techniques coupled with simplified analytical and computational solution methods to obtain approximate solutions, while independent prototyping exploration by the student with the tutorial promoted interactive experiential learning.

Some of the considerations behind the selection and implementation of these two particular tools as the software of choice for this class included: (1) the accessibility of the software within the class; (2) the ease of learning (within a semester or less); and (3) the unified modeling environment provided by these tools. Pre-constructed virtual models were also made available to the entire class to facilitate further exploration of many of these concepts on an individual basis by "Virtual Experimentation".

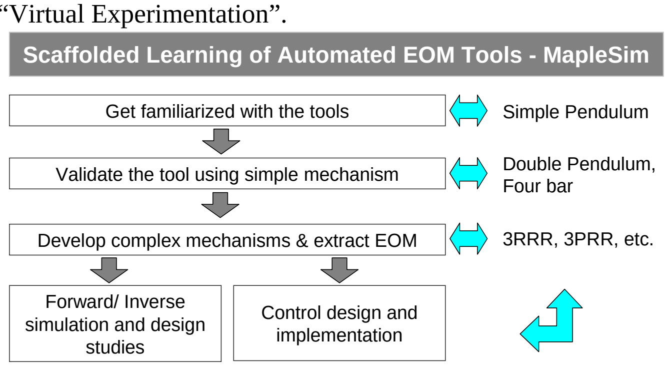


Fig. 3: The tutorial is intended for scaffolded learning of the tool (MapleSim) with examples range from simple pendulum to complex planar parallel mechanism, complementing the course lectures.

As shown in Fig. 3, in the *first* phase, the student begins with a series of simple case studies that are intended to familiarize the student with some of the basic functions in the MapleSim environment with the help of the examples and theory they learn in classroom. In the *second* phase, given some basic models of AMBS examples, students make advanced model in MapleSim and use what they have learnt in the previous steps to study the functional performance of these mechanisms (see Fig. 4). Finally, in the *third* phase, students use what they have learned in this tutorial to support their final design project of this course. The final project requires the student to use the software to explore different options in their designed model, such as interactively alter parameters and location of actuation, of their specific designed models and come up with the final design, which meets the specifications. These case-studies have a natural hierarchical staging in the form of increasing "problematic" components at subsequent levels as students gain mastery at the initial levels [8].

In addition, we also incorporated erroneous (but "intuitive") directives at several places within the tutorials. This is intended to force students to make common but undesirable mistakes and to then benefit from the experiential learning process. In doing so, our approach is closer to Linn's framework for scaffolded knowledge integration [6] which emphasizes the merits of such trial-and-error learning as well as the applicability of multiple equally-valid alternate approaches to problem solution. Ultimately, the principal desired outcome of these tutorials is to promote the development of *cognitive inquiry* within the student and accelerating their learning experience without increasing the lecture hours. At the same time the created framework also helps us address the more immediate goals of reinforcement of concepts being presented in the course by paralleling the course material in the case-studies.

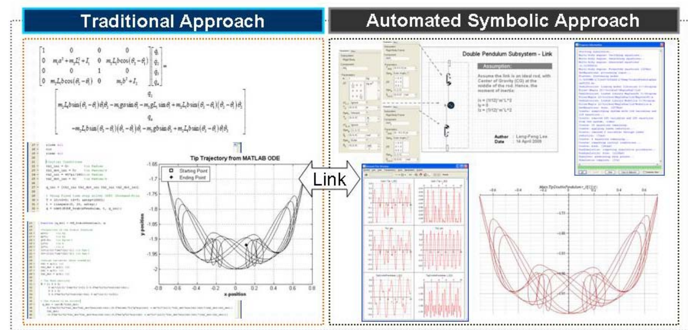


Fig. 4: Comparison of forward dynamic simulation in traditional approach and automated symbolic approach, with double pendulum as an example.

CASE STUDY 1: SIMPLE PENDULUM

The first case study is a simple pendulum shown in Fig. 5(a). In this first tutorial, the goal is allow the student to (i) get familiarized with the interface – such as where to locate all the building blocks of a system, how to connect various blocks to create a system, how to change the properties of a rigid body, etc.; (ii) explore various features provided by MapleSim – such as how to set the initial conditions, how to perform forward dynamic simulation, how to plot the results, as well as viewing the 3D visualization. This serves as an example of the types of tutorials in the first phase. The tutorial is introduced as a problem statement:

> *"A single pendulum of length of L, as shown in Fig. 5(a), with a mass m is pulled back to reach an initial angle of θ_0 from the vertical reference line and then released from rest. Determine the velocity and the reaction force over the entire period of the mass."*

This problem was selected to be the first example both from the viewpoint of its simplicity as well as its familiarity to the students. We demonstrate the process of modeling and the solution first by the traditional analytical approach, and then demonstrate the same process in MapleSim.

In the analytical approach, we discuss the following staged solution process:

(1) Idealize the problem by making appropriate assumptions, such as its inertia is like that of an idealized rod.
(2) Draw the free body diagram of the simplified model.
(3) Develop the appropriate governing equation of motions (EOM) using Newton's Laws of Motion as well as using Lagrangian method.
(4) Solve the EOM to obtain the desired solution.

In the MapleSim approach, the students are required to convert the simplified model into a Block based model as shown in Fig. 5(b). The tutorial shows the students how to create the parts and how the mechanism is assembled into the required initial configuration in MapleSim environment. The model can now be simulated in order to visualize its motion and extract the time history of important parameters.

However, it is important to motivate the students to exercise good engineering judgment while analyzing many of these virtual models. For example, since the simulations are done numerically, students were required to use their engineering knowledge to detect any inconsistencies, if present. In this pendulum example, the expected velocity vs angle graph is well understood by even the novice students. So, we check if the graph attains zero at maximum theta values and maximum value at an equilibrium value of theta (0 deg). Another example would be the x-y plot of the endpoint to lie on a circular path. We can do many such checks based on the model being analyzed.

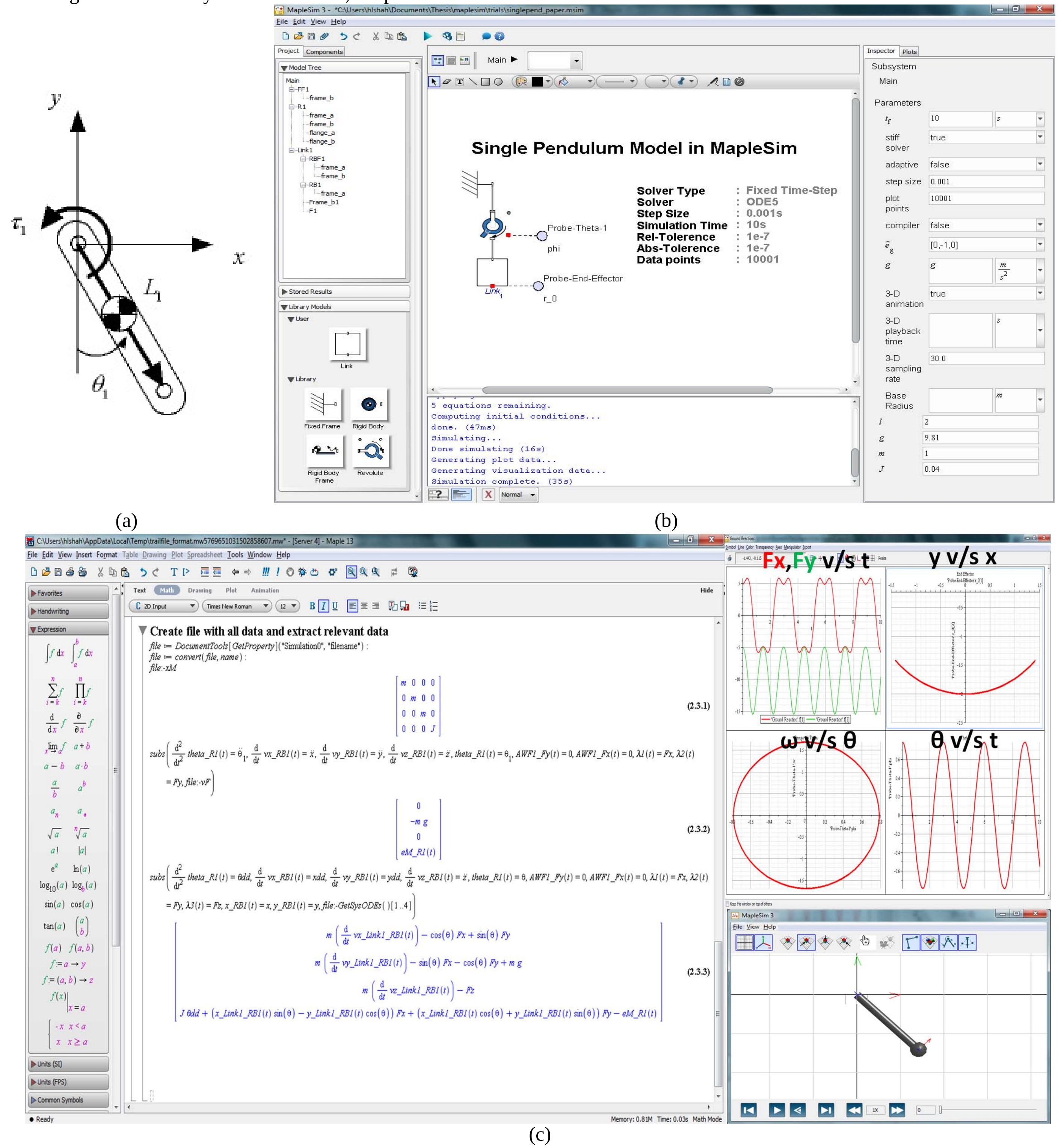


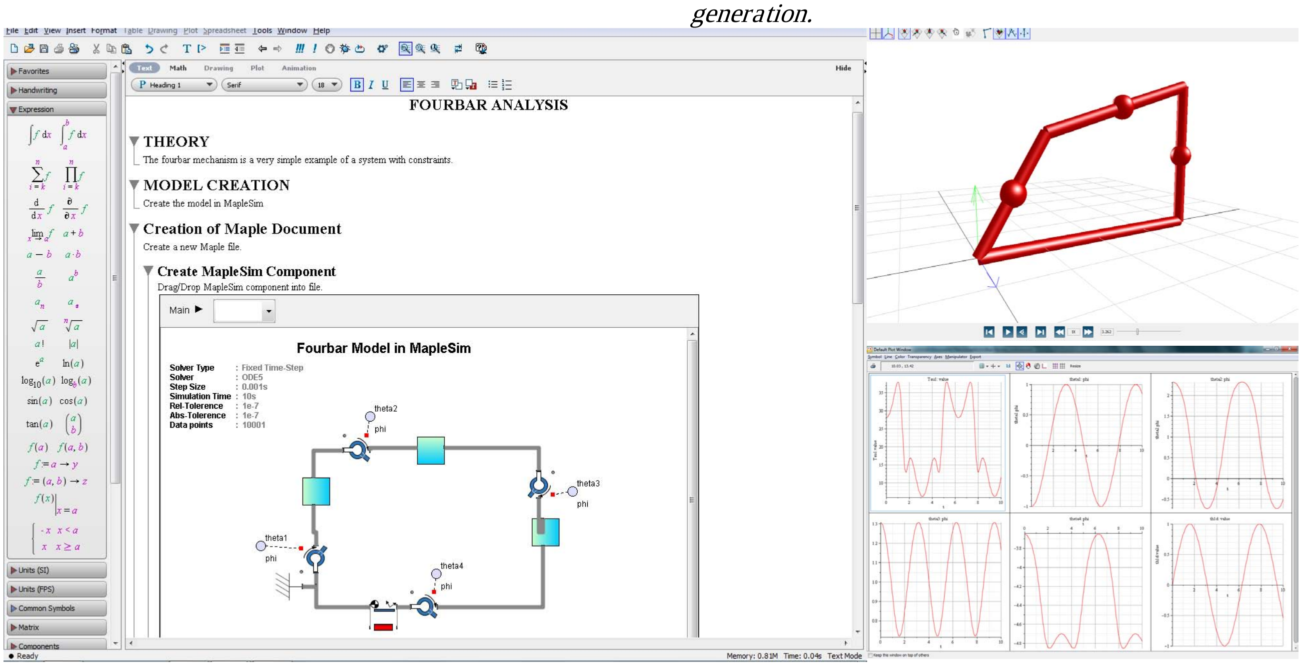
Fig. 5: (a) Simple pendulum problem; (b) modeled in MapleSim environment; (c) generated EOMs, plot of results from forward dynamic simulation, and visualization of the pendulum in motion.

In particular, we adopted the approach of first creating scenarios that caused errors, and then working the students through the process of resolving these errors. For example, by using a link as the model for the pendulum, the MapleSim model has a mass at the end of the link; while in most cases, the center of mass is assumed at the center of the link. The tutorial guided the students to recognize this difference, and work their way to simulate how it affected the model and hence understand it better.

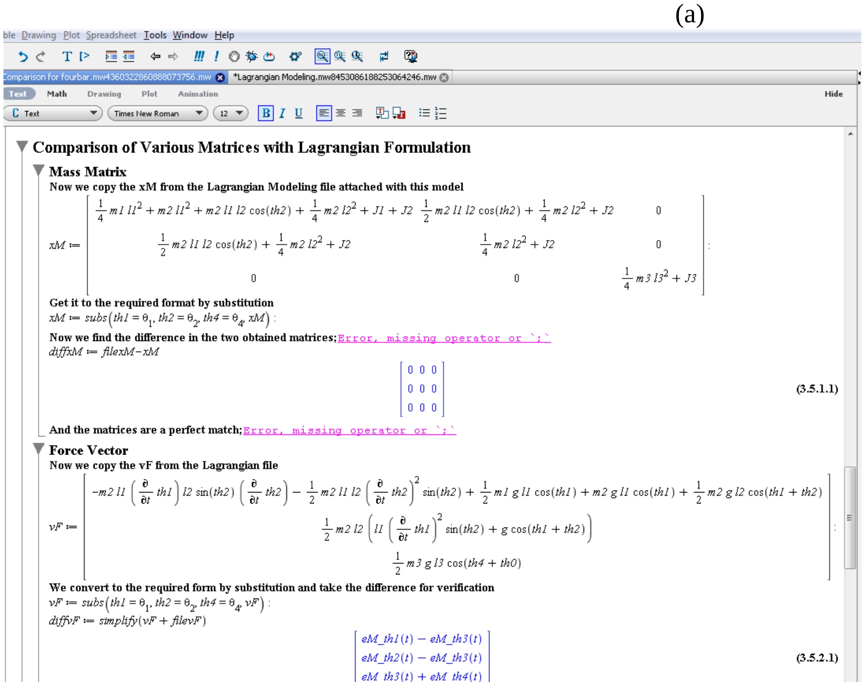
CASE STUDY 2: PLANAR FOUR-BAR MECHANISM

In the second phase, we examine more complicated examples of different planar mechanisms (double pendulum and fourbar, which are usually assigned to student as homework). The fourbar introduces a constrained mechanical system with dynamics couplings among the linkages. A fourbar, as shown in Figure 5(a), with the problem statement described as the following:

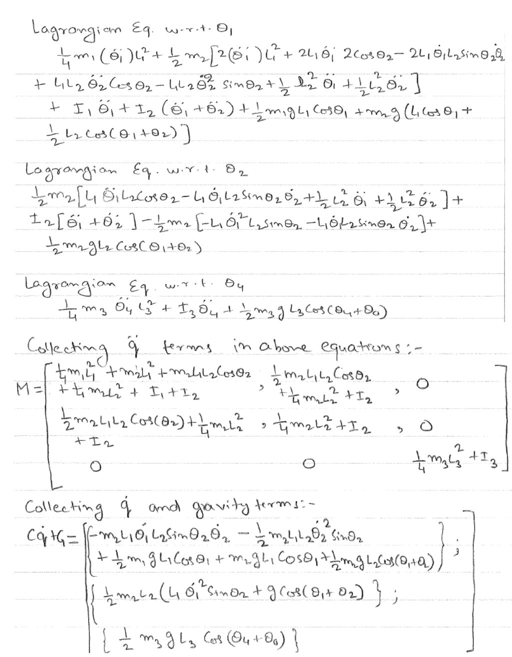
Given a fourbar mechanism, formulate the dynamic equations of motion (EOMs) using Lagrangian method and compare the results with MapleSim automated EOMs generation.



(a)



(c)



(b)

Fig. 6: (a) Fourbar mechanism modeled in MapleSim and its corresponding forward dynamic analysis; (b) analytical EOMs generated by using Lagrangian method (selectively shown only the EOMs and Mass matrix); and compare the EOMs with (c) EOMs generated using MapleSim.

Perform forward dynamic simulation of the fourbar system under the effect of gravity. Compute the required torque for input angle to follow a sine wave.

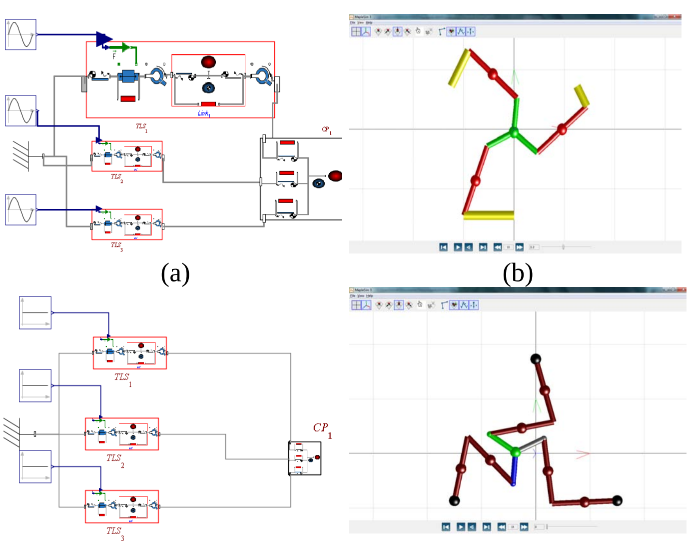
Traditionally the analytic approach is used to compute torque profile for a given motion profile. Using MapleSim, we reinforce the understanding of the idea by having the students visualize the same concept in the MapleSim environment. It is also possible to cross check results obtained from different formulations against one another. Eg. Calculation of constraint forces using EOMs in hand coded simulations and comparison with answers obtained from MapleSim.

The torque analysis can also be easily demonstrated and reinforced by a combination of both approaches. For a given set of parameters, the analytical approach can yield the torque profile easily by Euler Lagrange analysis. The same analysis this can be perform in the MapleSim to improve overall understanding.

Similarly, at each stage of the kinematic and dynamic analysis, the correspondence between the analytical approach and the MapleSim approaches are also emphasized at the same place. Finally, it is also important to have students aware of the trade-off between the symbolic equation generation and corresponding computation power by the tools and the accuracy of the results.

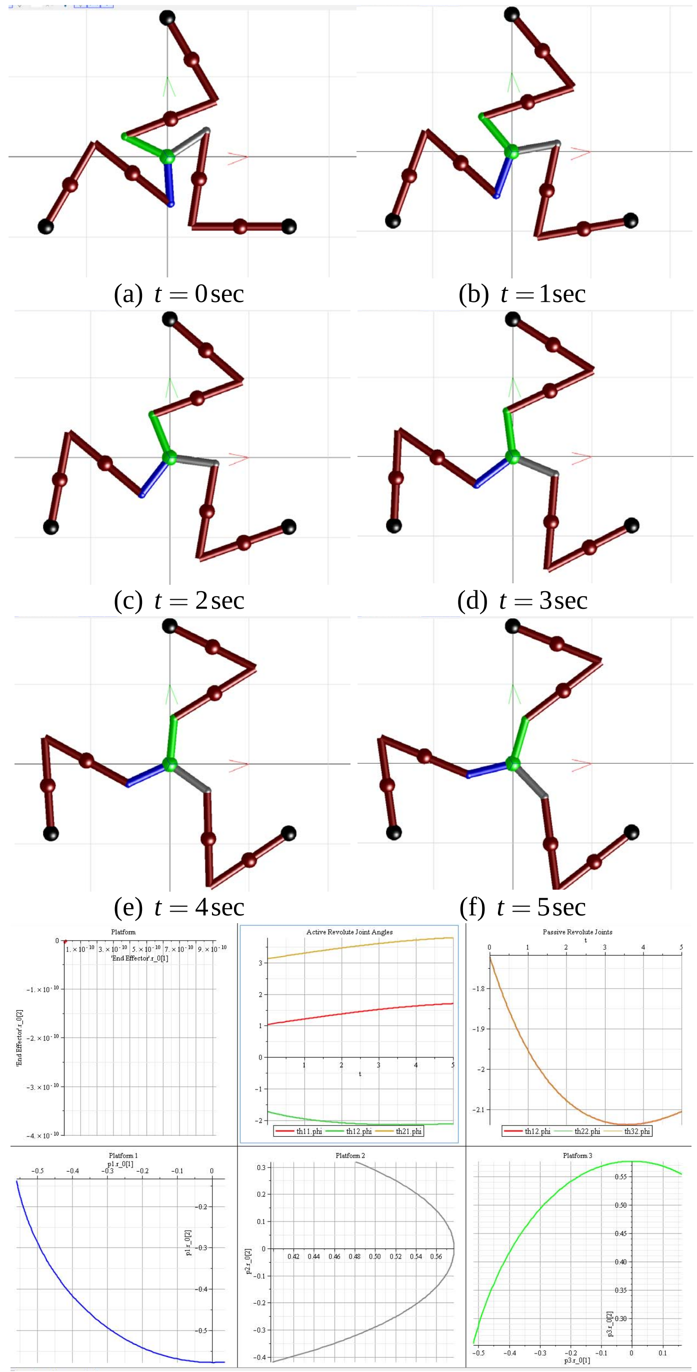
CASE STUDY 3: COMPLEX PLANAR PARALLEL MANIPULATOR

In the final phase, which assigned as a course final project, each students were assigned variants of planar parallel mechanism: 3PRR, 3RRR, 3RPR, to name a few. These systems introduce multiple loop constraints as compared to a fourbar system. They were required to formulate the EOMs of the given system, both analytically (using Lagrangian method) and using MapleSim, After they formulated the EOMs, they can then proceed to design controller to allow the mechanism to perform tasks such as trajectory tracking. They also study the performance measure of the system such as workspace and manipulability.



(a) (b) (c) (d)

Fig. 7: MapleSim model of (a) 3PRR and (b) 3RRR manipulator and their corresponding visualization shown in (c) and (d).



(a) $t=0\,\text{sec}$ (b) $t=1\,\text{sec}$ (c) $t=2\,\text{sec}$ (d) $t=3\,\text{sec}$ (e) $t=4\,\text{sec}$ (f) $t=5\,\text{sec}$ (g)

Fig. 8: (a)-(f) Forward dynamic simulation of a 3RRR manipulator under constant torque applied at the active joints for t=0 to t=5 sec; and (g) plot of end-effector position and active and passive joint angles, as well as locations of point of interest on the manipulator.

Fig. 7 (a) and (b) show one such example – a 3PRR parallel manipulator. The students were required to first solve the forward and inverse kinematics of the system, followed by formulating the EOMs analytically using Lagrangian method. Parallel to this work, student also creates the constrained multibody system in MapleSim, followed by generating EOMs symbolically. Fig. 9(a) and (b) shows the EOMs generated by (a) analytical approach and compare the result with (b) automated generated by MapleSim. Student would compare them and cross-check the EOMs. Similarly, a 3RRR case is shown in Fig. 7(c) and (d). Using this model, student were able to visually see how to the manipulator end effector move, if a constant torque of $-0.5N/m$ is applied at the three active joints for 5 seconds (shown in Fig. 8(a)-(f)). Various joint angles can also be plotted for parametrically studying the range of motion (workspace) of the manipulator, as shown in Fig. 8(g). Through this work, students were able to (i) gain deeper understanding of the analytical modeling process (by cross-checking their answer with one generated from MapleSim), and (ii) gain insights to the accuracy of such automated EOMs generation tools.

DISCUSSION AND CONCLUDING REMARKS

Based on the experience from the first offering of these tutorials, we note that the students had some initial difficulty but most were able to eventually effectively create virtual multibody models in MapleSim, generate EOMs automatically, compare the result with their analytical counterparts and subsequently verify the accuracy of the EOMs.

Most of the students were able to complete and check against the Kinematics simulation. This was mainly because the dynamic equations were very cumbersome to calculate manually and generally contained errors. Detailed symbolic manipulation could not be explored due to lack of tutorials explaining how equations could be extracted and interpreted. This is a shortcoming that our proposed tutorials seek to address.

The level of detail in the step-by-step instructions is a factor that we will be investigating further in future work. In some of this future work, we plan to gradually increase the number and complexity of intermediate “mini-projects” to permit the students to get hands-on practice in engineering problems with different levels of complexity. Video recordings of these tutorials were also created and available to student. These videos show step-by-step instructions on the creation of a MapleSim model, extracting EOMs, to performing various simulations.

From the instructional viewpoint this proved to be a viable vehicle for bridging the gap between the conventional classroom-based approaches for teaching mechanisms and robotics and an experiential approach. In terms of instructional support, very little was required to support the course (after the initial investment of effort and time in the tutorial).

Finally, a careful quantitative evaluation of the effectiveness of these tutorials is an important issue which still needs to be addressed and is being considered in future work.

ACKNOWLEDGMENTS

We gratefully acknowledge the support from the Research Foundation of State University of New York, National Science Foundation CAREER Award (IIS-0347653) and CNS-0751132.

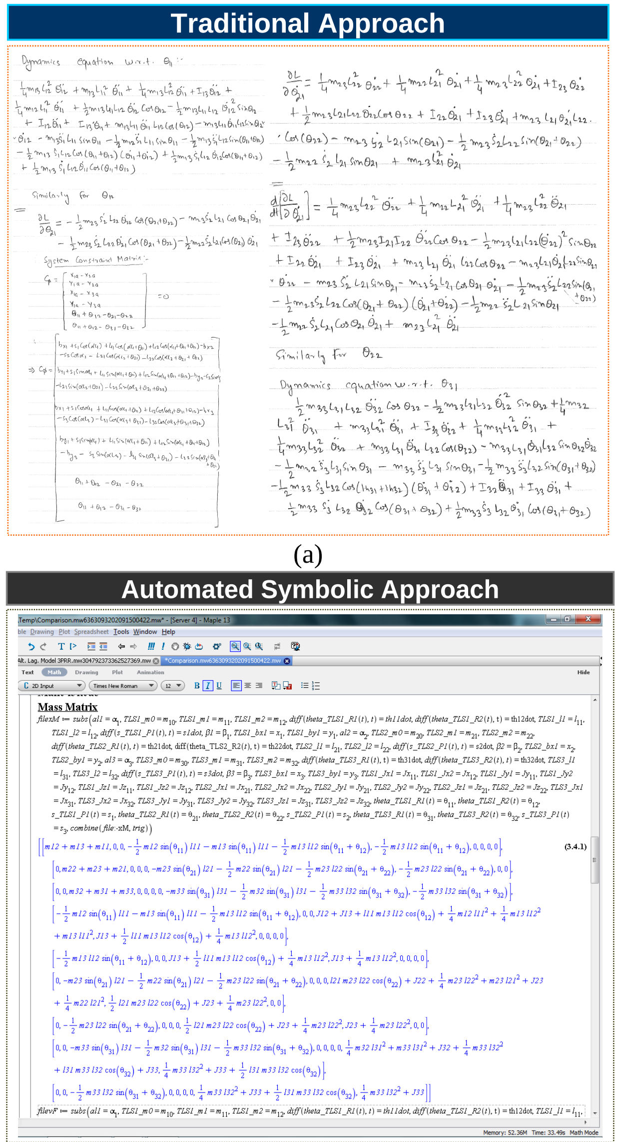


Fig. 9: (a) Analytical computation of EOMs of a 3PRR planar parallel mechanism and (b) Automated EOMs generation using MapleSim.

REFERENCES

[1] Ascher, U., and Petzold, L., 1998, Computer Methods for Ordinary Differential Equations and Differential Algebraic Equations, SIAM, Philadelphia, PA.
[2] Shabana, A. A., 2001, Computational Dynamics, Wiley, New York, NY.
[3] Waldron, K. J., and Kinzel, G. L., 2003, Kinematics, Dynamics, and Design of Machinery, Wiley, New York, NY.
[4] Maplesoft, 2009, "Maplesim," Waterloo, Ontario, Canada.
[5] Adams, G., and Jong, I., 1997, "Using Matlab to Animate the Generation of a Space Centrode in Kinematics," Proc. ASEE 1997 Annual Conference, Milwaukee, WI.
[6] Linn, M., 1995, "Designing Computer Environments for Engineering and Computer Science: The Scaffolded Knowledge Integration Framework," Journal of Science Education and Technology, 4(pp. 103-106.
[7] Shah, H., 2010, Mae 413/513 Maplesim Tutorials, http://sites.google.com/site/maplesimtutorial/
[8] Hiebert, J., Carpenter, T. P., Fennema, E., Fuson, K., Human, P., Murray, H., Olivier, A., and Wearne, D., 1996, "Problem Solving as a Basis for Reform in Curriculum and Instruction: The Case of Mathematics," Educational Researcher, 25(4), pp. 12-21.