

## REAL-TIME DYNAMIC MODELING AND VIRTUAL PROTOTYPE SIMULATION OF MECHANICAL SYSTEMS

**Pinhas BEN-TZVI**

Robotics and Mechatronics Lab  
Mechanical And Aerospace Engineering  
The George Washington University  
Washington, DC, USA  
[bentzvi@gwu.edu](mailto:bentzvi@gwu.edu)

**Mikhail ZHUK**

Department of Mechanical Engineering  
The University of California Irvine  
Irvine, CA, USA  
[mzhuk@uci.edu](mailto:mzhuk@uci.edu)

**ABSTRACT:** An alternative approach is developed towards the prototyping and testing phases during the development cycle of complex mechanical systems. Using this approach, virtual real-time dynamic simulation and testing circumvent costly physical prototyping in the early stages of the development. With this approach, a virtual model of the system is created using CAD based software, imported, and constrained within Automatic Dynamic Analysis Mechanical Systems (ADAMS) software in order to run an initial dynamic analysis. The virtual model is further constrained through a control system using Matlab/Simulink software. Once a detailed control system is developed in Matlab/Simulink, ADAMS Controls software is interlinked with the Simulink control systems in order to run real-time dynamic simulations using external signal inputs emanating from another subsystem. As a case study using this approach, a CAD model of a Hybrid Mechanism Mobile Robot (HMMR) was developed in Solidworks and imported into ADAMS View software. A closed loop control system was developed using Matlab/Simulink software and interconnected with ADAMS Controls in order to conduct a real-time dynamic simulation. The real-time simulation was performed based on output files received from the stereo vision (on-board the HMMR) based program in order to test the parameters of the motor control. Using this approach enabled us to test the main control algorithms before final part selections were made and manufacturing of a prototype was underway. The simulations resulted in diagnosis, testing and modification of inadequate command constants and functions required to efficiently perform autonomous obstacle climbing.

**Keywords:** Dynamic Model, Real-time Simulation, Virtual Prototyping, MSC Adams Controls

### I- INTRODUCTION

With fierce competition across the industry, companies are focusing on developing alternative approaches in order to concurrently reduce overall development time and costs. Unfortunately, along with the advent of inexpensive high speed computing, the complexities of mechanical systems have evolved to become increasingly interlinked with electrical and software elements. With the synergy and intricate relationships among the three main branches behind mechatronic systems, the development of alternative approaches become necessary in order to both reduce development time during the prototyping and testing phases of the product development cycle as well as improve system functions between interdependent fields [1].

Overall, a basic simplified design cycle can be reduced to nine steps (as seen in Figure 1). One major element during the cycle consists of the testing and reevaluation of the design prototype as this establishes the validity of the theoretical model's capability to withstand its designed application as well as physically confirms the validity of the new concepts it is based on. Although rapid prototyping has dominated this area of the development phase, current rapid prototyping technology is far from ideal. Rapid prototyping can be classified into three main types according to the materials used: powder, resin, and laminated sheet based [2]. Despite the advantages of rapid prototyping when compared to the issues in constructing a scaled model, the accuracy and quality of using newer techniques depends greatly on a number of process parameters, and requires a significant degree of expertise in order to create a more accurate model [3].

However, experimental study of physical prototypes is quite costly and time consuming [4]. In an attempt to shorten the conventional product development process, virtual prototyping and real-time dynamic simulation are

implemented in order to verify product design and optimize design parameters. Although virtual modeling software such as Solidworks, Pro-Engineer and other CAD based software are readily available, they are normally not capable of generating analyses beyond static or basic dynamics. In order to effectively simulate the behavior in real-time, the virtual model must take into account the broader range of interdependent aspects, including the electrical and software base. This is done through co-simulation, allowing for concurrent component simulation and the simultaneous exchange of information by specialized simulation tools for each subsystem. For example, the solid model is created using Solidworks, the dynamic simulation is conducted through ADAMS View/Controls, the control system is simulated through Matlab Simulink and the command signal is generated from a C++ compiler. Using this simulation modeling method the resulting test data on both the mechanical and the electrical elements are attained before any physical prototype is developed, allowing for a shortened transition time to market, as well as knowledge required for earlier testing of control and programming aspects.

## II- HMMR BACKGROUND

As mobile robot platforms advance, they are implemented in high risk field operations for law enforcement, military applications, hazardous cleanup, and planetary exploration. The Hybrid Mechanism Mobile Robot (HMMR) as shown in Figure 2 is an adaptation of a mobile robot design based on hybridization of the mobile platform and manipulator arm as one entity for robot locomotion as well as manipulation [5],[9]. Due to the symmetric nature of the design, the platform can continue to the destination even if the robot is inverted, allowing it to utilize Link 2 in order to

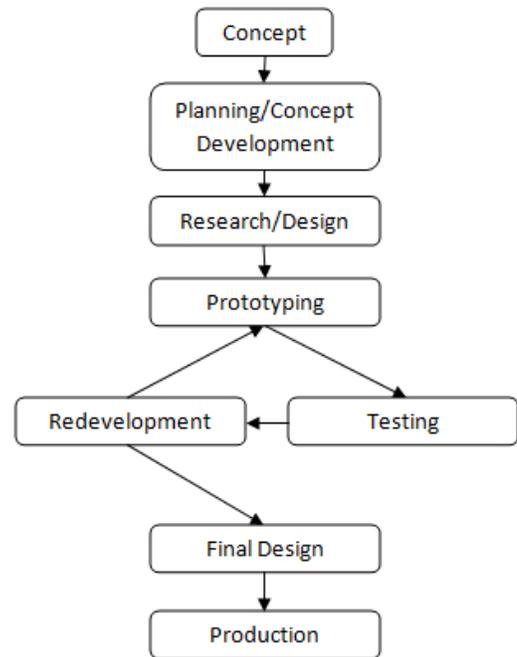


Figure 1: Simplified Design Cycle

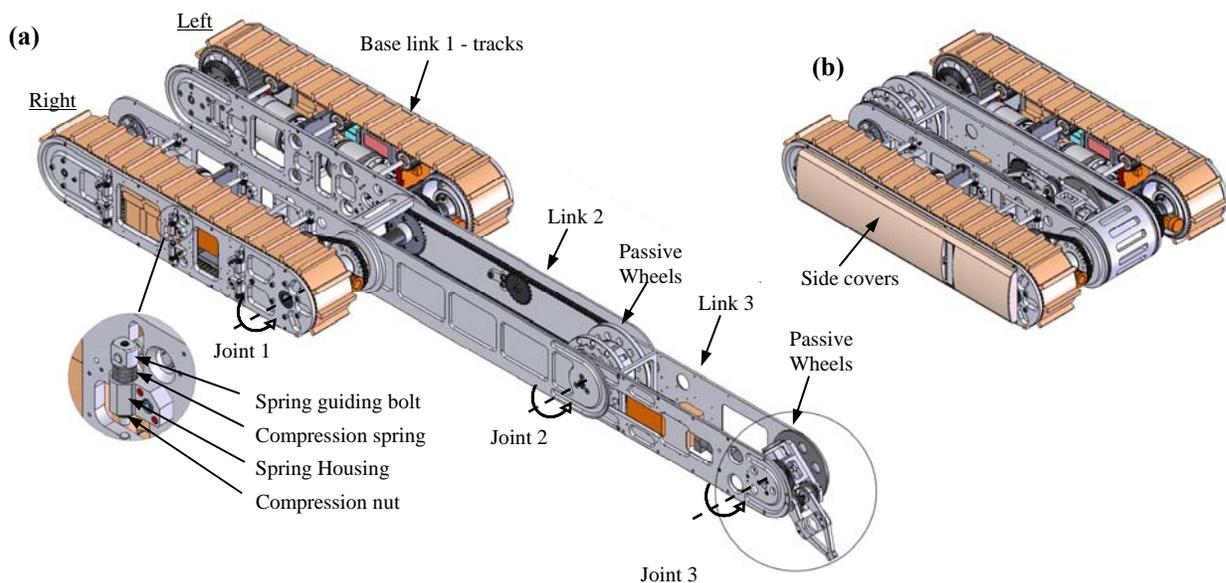


Figure 2: HMMR - (a) open configuration mode; (b) Stowed-links configuration mode

stabilize the platform and climb on or descend from obstacles. The platform includes two identical and parallel base link 1 tracks (left & right), link 2, link 3, end-effector (EE) and passive wheels. To support the symmetric design, all the links are nested into one another. Link 2 is connected between the two base link tracks via joint 1 (Fig. 2(a)). Passive wheels are inserted between links 2 and 3 and connected via joint 2 and another passive wheel is inserted between link 3 and the EE via joint 3 (Fig. 2(a)). The passive wheels are used to support links 2 and 3 when used for various configuration modes of locomotion/traction. Link 2, link 3 and the EE are connected through revolute joints and are able to provide continuous 360° rotation and can be deployed separately or together from either side of the platform. To prevent immobilization of the platform during a flip-over scenario, rounded and pliable covers are attached to the sides of the platform as shown in Fig. 1(b). The design also includes a built-in dual-operation track tension and suspension mechanism that accounts for the symmetric nature of the design and is also used to absorb some of the energy resulting from falling or flipping. Readers are referred to references [5] & [6] for details on the design development, analyses and relevant technical characteristics and features of the system (i.e., physical dimensions, weight, torques and comparisons to other mobile robot systems).

Unlike the original mobile robot, the HMMR is designed with autonomous capability. In order to further extend its range of motion capabilities, the control architecture has been redesigned based on an embedded on-board wireless communication network between the robot's links and modules. Using this modular control approach, the functionality of the mobile robot is increased by allowing the continuous rotation of each link. In addition, the HMMR is equipped with both stereo vision system and pitch-actuated LIDAR systems in order to identify and map out the surrounding 3D environment via use of vision based algorithms. Using the autonomous obstacle negotiation function, the HMMR will autonomously navigate through rough terrain environments by dividing the terrain into a sequence of simple models and recognizing each obstacle using surface detection. Each of the divided terrain sequences will be navigated through by summoning one of seven built in control libraries to manipulate each of the four drive motors. Due to the complexity of this mechatronic system, co-simulation is required in order take into consideration the interdependency between each module and control algorithm in order to create an accurate virtual model.

### III- VIRTUAL PROTOTYPE AND SIMULATION

When designing a mechanical system such as this hybrid robot, it is required to understand how various components interact as well as what forces/constraints those components

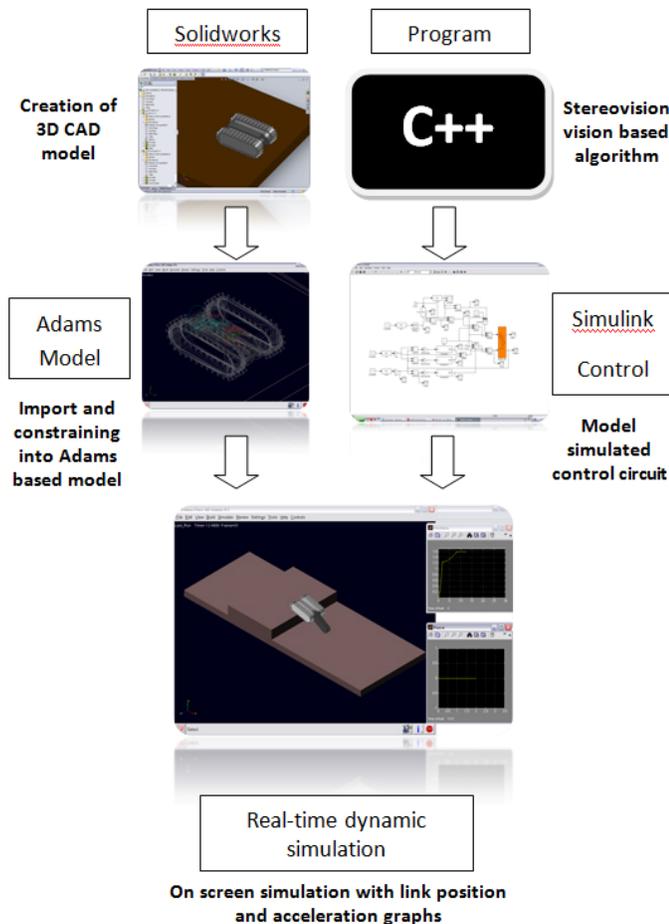
generate during operation. ADAMS is commercial motion simulation software for analyzing the behavior of complex mechanical systems. It allows testing virtual prototypes and optimizing designs for performance, without having to build and test physical prototypes. This dramatically reduces product development time and cost. Once the design parts are generated with a CAD Software, they can be either imported to ADAMS directly or used to create first a digital mock-up (fitting the parts by assembling), and then import the design assembly to create the virtual prototype for functional tests.

In order to study the functionality of the HMMR, a virtual prototype [7] is required to represent an accurate assembly of all subsystems, including those associated with the control of informatic and module signals. This is accomplished through the use of co-simulation involving MSC Adams, Matlab Simulink, our C++ algorithm, and the stereo vision signal (as seen in Figure 3) [8]. To test and fine tune the climbing library, the system uses input signals from the stereo vision in order to derive the height and distance of the nearest object and calculate the required position and link movement in each step using within the C++ algorithm. Once the step values are calculated the control system manipulates the four motors to perform the motion in each step. Due to the interdependency of each subsystem, the virtual model consists of three main parts: a representation of the physical mechanical model, a closed loop control system, and a calculation of control values derived from the stereo vision signal. To create a physical representation, the 3D mechanical model created in Solidworks was exported to MSC Adams dynamic motion software, where all joints, connections, contact forces, mass distribution, and inertial properties are accounted for and constrained. Once the representation of the physical system is fully defined, a closed loop control system is required in order to standardize and control the input signal to each motor. Using a combination of MSC Adams Controls and MATLAB Simulink, a controls plant is exported from Adams into Simulink with input signals for both the link motor and drive motor as well as output signals for the x-axis and angle position. A control system then is devised after taking into account which of the seven built-in control libraries is being replicated. In this case study, the step climbing function consists of the following steps:

- (i) locomotion and positioning,
- (ii) manipulation of the link to position the platform onto the step,
- (iii) locomotion in order to move the platform completely onto the step, and finally
- (iv) manipulation of the link to its initial position.

The output of the C++ algorithm calculating the required position and angle values is imported into Simulink through a mutual server. Once the virtual prototype is set up, a real-time dynamic simulation is performed using stereo vision and the physical obstacles represented in the 3D mechanical model. The resulting data obtained from the simulation consists of HMMR position, link angle, and torque applied by each

motor with respect to time. Through the integration of the obtained data along with the visual representation, the control algorithms are validated and optimized to meet specific parameters of the motor controls.

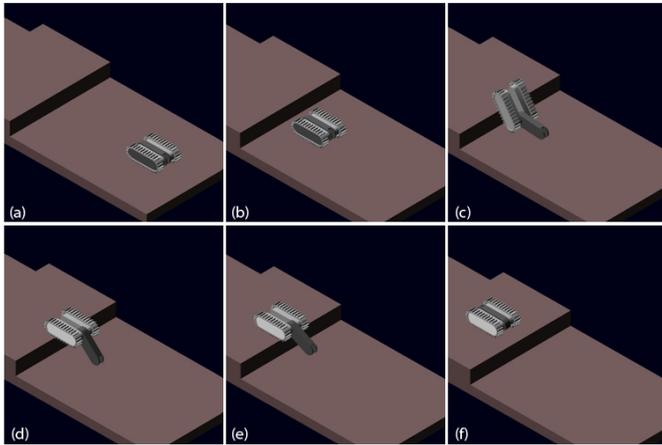


**Figure 3: Virtual Prototype Design Process**

#### IV- SIMULATION RESULTS AND DISCUSSION

The real-time dynamic simulation was performed for the purpose of studying the robot’s functionality and validating the program output parameters controlling each motor for the desired movement. The climbing library consisted of four main substeps as can be seen in Figure 4. Using the output distance and angle position obtained from simulation data, the minimum and maximum command ranges required for the HMMR to complete each function was determined while allowing for optimized gain values for the C++ algorithm. The current model is limited in a number of aspects that will be improved upon in future simulations. First, the current model contains fixed tracks which limit the accuracy of the transversed movement. This can be corrected as a remodeled version will be created using MSC Adams Track Toolkit.

Second, due to the simplicity of this model, there is only one link limiting the motion capabilities. This too will be addressed in the subsequent version of the model that will be based off the detailed CAD-based HMMR model. The third restricting factor of this simulation is the predetermined fixed location and size of the obstacle in the model. Though the stereo vision is able to detect the distance between robot and the object, object width and height, we are not currently able to automatically generate the obstacle in MSC Adams. The fourth restricting factor consists of computer resources. In order to run the real time dynamic simulation with on screen updates, computer processing power becomes an issue. As the communication between Matlab Simulink and MSC Adams occurs every 0.005 seconds the model simulation time varies with complexity of the calculations required between contacting bodies causing inconsistencies between real time and simulation time. Unfortunately, this cannot be easily bypassed by manipulating the communication rate between Simulink and Adams as this creates disjunct instantaneous motions effecting the overall dynamic response of the model. One possible solution to this problem that has yet to be looked in to is possibly the optimization of the Simulink control system. The fifth restricting factor is the simulink computational use of the imported C++ algorithm output. Although the calculated output file representing the desired angle location and the desired position can be updated nearly instantaneously as the HMMR is moving towards the obstacle, the use of an output signal in the form of an updated text file through a mutual server instead of a physical output signal greatly restricts the amount of data that can be imported into simulink. At the start of the Simulink simulation, Simulink imports the given parameters from the text file once and continues using those initial parameters throughout the entire simulation. As the control system is able to use those initial parameters to guide the HMMR to each position, more than one single set of parameters are not required, but do not give an accurate representation of the functionality of the stereo vision system. This can be overcome by either wirelessly linking the two computers through the use of RF modules or physically connecting the two computers using serial cable in order to create a physical output/input signal. With a physical output signal Simulink is able to update at a specified rate throughout the entire simulation. Using this method would allow for more complex simulations that could mimic rough environments in order to more accurately test obstacle recognition as well as multiple sequential control library use. Overall, a new virtual prototype is required to be developed that will simultaneously link and test the wireless modules.



**Figure 4: (a) Initial Position; (b) Positioning: Move forward to the initial climbing position; (c) Rotation of link 1 by angle  $\theta$ ; (d) Move forward to surmount the obstacle; (e) Close link 1; (f) Stop**

## V- CONCLUSIONS

This paper presents an alternative approach towards the prototyping and testing phases during the development lifecycle of complex mechanical systems through the use of real-time co-simulation. A virtual prototype of the control system was developed through the use of MSC Adams interlinked with MATLAB Simulink in order to test the output commands of the specified control library. Using this approach has considerably reduced development time and costs, as well as aided in defining the optimal parameters for each control step.

The contribution of this research is a prescribed method for virtual prototyping through the use of MSC ADAMS, Matlab/Simulink and Solidworks or any other computer aided design based software. The proposed method allows real-time dynamic simulation that significantly help with effectively selecting system components (motors, sensors, etc.), and perform control program modifications, through external input sources during the earlier stages of system development before prototyping is possible or relatively costly.

## ACKNOWLEDGEMENTS

This work is supported by Defense Advanced Research Projects Agency (DARPA) under grant number R0011-09-1-0049. We also would like to acknowledge the support provided by the DARPA Program Manager, Ms. Melanie Dumas.

## REFERENCES

- [1] Ferretti, G., Magnani, G., Rocco, P., 2004, "Virtual Prototyping of Mechatronic Systems", *Annual Reviews in Control*, 28, pp. 4245 – 206.
- [2] Pham, D.T., Gault, R.S, 1997, "A Comparison of Rapid Prototyping Technologies", *International Journal of Machine Tools & Manufactureing*, 38, pp. 1257 – 1287.
- [3] Choi S.G., Chann A.M.M, 2003, "A layer-based virtual prototyping system for product development", *Computers in Industry*, 51, pp. 237 – 256.
- [4] Baxter D., Gao, J., Case K., Harding, J., Young, B., Dani, S., Cochrane, S., 2007, "An Engineering Design Knowledge Reuse Methodology Using Process Modeling", *Res Eng Desig*, 18, pp. 37– 48.
- [5] Ben-Tzvi, P., Goldenberg, A.A., Zu, J.W., 2008, "Design and Analysis of a Hybrid Mobile Robot Mechanism with Compounded Locomotion and Manipulation Capability", *Transactions of the ASME, Journal of Mechanical Design*, 130, pp. 1–13.
- [6] Ben-Tzvi, P., Goldenberg, A.A., and Zu, J.W., May 2008, "Design, Simulations and Optimization of a Tracked Mobile Robot Manipulator with Hybrid Locomotion and Manipulation Capabilities", *Proc. of the 2008 IEEE Int. Conf. on Robotics and Automation*, Pasadena, California, pp. 2307–2312.
- [7] Ben-Tzvi, P., Raoufi, C., Goldenberg, A.A., and Zu, J.W., October 2007, "Virtual Prototype Development and Simulations of a Tracked Hybrid Mobile Robot", *Proceedings of MSC. Software 2007 Virtual Product Development Conference*, Detroit, Michigan.
- [8] Zhang, H., Wang, H., Chen, D., Zacharewicz, G., 2010, "A Model-Driven Approach to Multidisciplinary Collaborative Simulation for Virtual Product Development", *Adv. Eng. Informatics*, 24(2), pp. 167-179.
- [9] Ben-Tzvi, P., May 2010, "Experimental Validation and Field Performance Metrics of a Hybrid Mobile Robot Mechanism", *Journal of Field Robotics*, Vol. 27, Issue 3, pp. 250-267.