STM JOURNALS

International Journal of
Mechanical Dynamics and
Systems Analysis

Volume 3, Issue 1, 2025
January–June
DOI (Journal): 10.37591/IJMDSA

https://journals.stmjournals.com/ijmdsa

Review IJMDSA

Dynamic Modeling and Simulation of Multi-Body Mechanical Systems: A Comprehensive Review of Methods, Tools, and Applications

Shubham Mishra[1,*]

Abstract

The dynamic modeling and simulation of multi-body mechanical systems (MBS) form a cornerstone in modern mechanical engineering, enabling in-depth analysis of the kinematic and kinetic behaviors of interconnected rigid and flexible components. MBS are foundational to a range of critical applications, from automotive suspensions and aerospace mechanisms to robotics and biomechanical structures. As system complexity and performance requirements increase, accurate and scalable modeling techniques are essential for both design validation and performance optimization. This review systematically explores fundamental modeling formulations, including Newton–Euler, Lagrangian, and Kane's methods, as well as strategies for constraint stabilization and numerical integration. A detailed examination of industry-standard and open-source simulation platforms—such as Automated Dynamic Analysis of Mechanical Systems (ADAMS), Simpack, RecurDyn, Project Chrono, and MBsysC—is presented, with a focus on capabilities for co-simulation, flexible body dynamics, and control system integration. The paper analyzes domain-specific MBS applications and addresses emerging paradigms such as AI-based modeling, real-time digital twins, and multiphysics co-simulation. By bridging foundational theory with modern software tools and outlining future research trajectories, this article aims to serve as a consolidated reference for engineers, researchers, and system analysts engaged in mechanical dynamics and system-level simulation. By bridging foundational theory with modern software tools and outlining future research trajectories, this article aims to serve as a consolidated reference for engineers, researchers, and system analysts engaged in mechanical dynamics and system-level simulation. Emphasis is also placed on the importance of interoperability between simulation tools and control design environments, scalable model reduction techniques, and the integration of sensor data for model calibration. The review concludes by identifying key challenges and opportunities in the development of next-generation MBS frameworks that are more adaptive, intelligent, and responsive to real-world operating conditions.

Keywords: Multi-body dynamics, dynamic system modeling, mechanical simulation, constraint handling, real-time simulation, numerical integration, co-simulation tools

***Author for Correspondence**
Shubham Mishra
E-mail: subhammishra7860@gmail.com

[1]Research Scholar, Department of Electrical Engineering, Jaipur Engineering College and Research Centre, Jaipur, Rajasthan, India

Received Date: March 22, 2025
Accepted Date: June 12, 2025
Published Date: June 24, 2025

Citation: Shubham Mishra. Dynamic Modeling and Simulation of Multi-Body Mechanical Systems: A Comprehensive Review of Methods, Tools, and Applications. International Journal of Mechanical Dynamics and Systems Analysis. 2025; 3(1): 35–42p.

INTRODUCTION

Multi-body mechanical systems (MBS) are integral to the modeling and analysis of real-world engineering applications where multiple rigid or flexible bodies interact through joints, constraints, and force elements. These systems are widely used in the design and optimization of vehicles, aerospace mechanisms, industrial machinery, robotic manipulators, and biomedical devices. The increasing complexity and performance demands of such systems necessitate advanced dynamic modeling techniques capable of accurately capturing their transient and steady-state behaviors under various operational conditions [1–3].

At the core of MBS dynamics is the ability to describe the motion of interconnected bodies that are subjected to external forces and internal constraints. These motions can be highly nonlinear owing to geometric constraints, frictional contacts, elastic deformations, and actuator effects. Therefore, a rigorous mathematical framework is essential for deriving the governing equations of motion. Classical formulations such as Newton–Euler equations provide a force-based perspective, whereas Lagrangian and Kane's methods offer energy-based and computationally efficient alternatives, respectively. Each approach has its own merits depending on the system structure, degrees of freedom, and nature of the constraints [4–6].

The accurate prediction of system behavior through dynamic simulations plays a pivotal role in modern engineering workflows. It enables virtual testing, failure prediction, and design optimization, often replacing or minimizing the need for expensive physical prototypes. For example, in automotive engineering, an MBS simulation is utilized to analyze suspension dynamics, crash behavior, and driveline performance. In robotics, it supports control design and trajectory planning. In aerospace systems, MBS models are crucial for analyzing deployment mechanisms and controlling surfaces under various flight regimes. Similarly, biomechanical studies use MBS to investigate human motion, joint loading, and prosthetic development [7, 8].

Over the past few decades, advancements in computational power have revolutionized MBS simulations. A wide array of commercial and open-source software tools currently exists, providing user-friendly interfaces and robust solvers for efficient model development and simulation. These tools often support co-simulation capabilities, allowing integration with control systems, finite element models, and fluid dynamics solvers. Furthermore, recent trends have shown increasing interest in data-driven modeling, real-time simulation, and digital twin technologies, which enhance predictive capabilities and enable continuous system monitoring [9, 10].

FORMULATIONS FOR DYNAMIC MODELING

Accurate dynamic modeling of MBS requires robust mathematical frameworks that describe the motion of interconnected bodies under various forces and constraints. Several formulations have been developed to derive the equations of motion for such systems, each offering distinct advantages based on the computational efficiency, system structure, and ease of implementation. This section reviews the three most widely used approaches, Newton–Euler, Lagrangian, and Kane's methods, along with the commonly adopted constraint stabilization techniques used to ensure accurate simulations [11, 12].

Newton–Euler Equations

The Newton–Euler formulation applies Newton's second law to each individual body in the system, treating the linear and angular motions separately. For a rigid body with mass m, position vector r, and moment of inertia I, the equations of motion can be expressed as:

Linear motion:

$$m \times \ddot{r} = F\_ext + F\_int$$

Rotational motion:

$$I \times \dot{\omega} + \omega \times (I \times \omega) = T\_ext + T\_int$$

Where, F and T represent the external and internal forces and torques, respectively, and ω is the angular velocity.

This formulation is intuitive and efficient for tree-structured and open-chain systems. However, for closed-loop systems or those with redundant constraints, the equations can become complex and require differential algebraic formulations.

Lagrangian Dynamics

Lagrangian dynamics is an energy-based approach that simplifies modeling when generalized coordinates are used. Lagrangian L is defined as the difference between the kinetic and potential energies:

$$L = T - V$$

The equations of motion are derived using:

$$d/dt\,(\partial L/\partial \dot{q}_i) - \partial L/\partial q_i = Q_i$$

Where, q_i are generalized coordinates, $\dot{q}_i$ their derivatives, and Q_i are generalized forces.

This method is well-suited for systems with complex constraints and fewer degrees of freedom, although it can become algebraically intensive for large systems.

Kane's Method

Kane's method combines Newtonian principles with the benefits of a reduced coordinate system. Instead of energy, it uses the generalized speeds u_i and partial velocities. The resulting dynamic equations are more compact.

$$\Sigma\ (\text{generalized active force}) - \Sigma\ (\text{generalized inertial force}) = 0.$$

This formulation avoids computing energy terms and is computationally efficient, making it ideal for symbolic computation and implementation in automated modeling tools.

Constraint Stabilization Techniques

In simulations involving closed kinematic loops or flexible bodies, constraints can drift over time owing to numerical integration errors. Constraint stabilization techniques were used to prevent this from occurring.

Baumgarte Stabilization modifies the second derivative of the constraint equation:

$$\ddot{\Phi} + 2\alpha \times \dot{\Phi} + \beta^2 \times \Phi = 0$$

Where, Φ is the constraint function, and α and β are the tuning parameters for the damping and stiffness-like correction.

Augmented Lagrangian Methods combine penalty functions and Lagrange multipliers to enforce constraints with improved numerical stability, particularly for stiff systems.

NUMERICAL METHODS FOR DYNAMIC SIMULATION

Numerical simulation is central to analyzing the dynamic response of MBS, as most real-world systems involve nonlinearities, constraints, and time-dependent forces that make closed-form solutions impractical. Once the equations of motion are derived, typically in the form of differential or differential algebraic equations (DAEs), they must be integrated over time using suitable numerical methods. The choice of the integration scheme significantly affects both the stability and accuracy of the simulation, especially in systems with stiff dynamics, high-frequency modes, or long simulation durations. This section reviews commonly used time integration techniques and discusses the critical trade-offs between stability, computational cost, and accuracy.

Time Integration Schemes

Time integration involves advancing the system state, typically positions, velocities, and accelerations, through discrete time steps. Common numerical schemes are categorized as explicit, implicit, or semi-implicit, depending on how they handle dependencies within the equations of motion.

Explicit Methods

Explicit schemes, such as the *Forward Euler* and *Runge-Kutta* methods, compute the system state at the next step using only the information from the current time step. For example, the Forward Euler method updates position and velocity using.

$v(t + \Delta t) = v(t) + a(t) \times \Delta t$

$x(t + \Delta t) = x(t) + v(t) \times \Delta t$

These methods are simple and computationally efficient but are only conditionally stable. They are best suited for systems with smooth dynamics and small-time step requirements.

Implicit Methods

Implicit methods, such as *Backward Euler* or the *Newmark-Beta* method, involve solving equations in which the future state depends on itself. For example, in the Backward Euler method:

$v(t + \Delta t) = v(t) + a(t + \Delta t) \times \Delta t$

$x(t + \Delta t) = x(t) + v(t + \Delta t) \times \Delta t$

These methods are unconditionally stable for linear systems and can handle stiff equations, rendering them suitable for systems with high-frequency dynamics or flexible bodies. However, they require iterative solvers and Jacobian evaluations, which increase computational costs.

Semi-Implicit Methods

These methods strike a balance between the simplicity of explicit methods and the robustness of the implicit schemes. For example, in the *Velocity Verlet* algorithm, the velocity is updated in two stages: partially using the current acceleration and then corrected using the new acceleration. They are widely used in physics-based simulations owing to their good energy conservation properties.

Stability and Accuracy Considerations

The selection of a time integration method must balance *stability, accuracy*, and *computational efficiency.* Stability refers to the ability of an algorithm to produce bounded solutions over time, which is particularly important for stiff or constrained systems. Accuracy refers to how closely the numerical solution approximates the true physical behavior.

- *Stiff Systems:* Systems with fast and slow dynamics (e.g., flexible bodies with high-frequency vibrations) require small time steps if explicit methods are used. Implicit integrators are preferred in such cases because of their superior stability.
- *Constraint Satisfaction:* In MBS with holonomic or nonholonomic constraints, drift from constraint manifolds can occur if integration is not handled carefully. Using stabilized implicit schemes or incorporating constraint correction steps (e.g., projection methods or penalty formulations) can maintain constraint fidelity.
- *Energy Conservation:* Some applications, such as long-term orbital simulations or biomechanical gait analysis, demand that numerical methods conserve mechanical energy. Symplectic integrators and energy-momentum methods are suitable choices.
- *Adaptive Time Stepping:* In dynamic systems with rapid transitions (e.g., contact events and impacts), adaptive time stepping improves efficiency by varying the step size based on the estimated error or stiffness.

Ultimately, the best integration scheme depends on the characteristics of the system being simulated, the level of model fidelity required, and available computational resources. Commercial MBS software often provides hybrid solvers that automatically switch between methods or adjust parameters based on the runtime conditions to maintain stability and performance.

SIMULATION SOFTWARE FOR MULTI-BODY SYSTEMS

Advancements in computing technology have significantly expanded the scope and capability of simulation tools used to model MBS. These software platforms allow engineers and researchers to construct complex system models, perform dynamic simulations, and analyze the results with high accuracy and computational efficiency. The selection of the simulation software depends on various factors, including the nature of the system (rigid or flexible), type of analysis (static, dynamic, or control), user interface preferences, and integration requirements with other tools. This section reviews both the commercial and open-source software platforms commonly employed for MBS simulations.

Commercial Platforms

Commercial software packages are widely used in the industry because of their advanced features, user-friendly interfaces, robust solver engines, and technical support. These tools often provide extensive libraries of mechanical components, built-in analysis functions, and integration with computer-aided design (CAD) and control design environments.

- *ADAMS (MSC Software):* Automated Dynamic Analysis of Mechanical Systems (ADAMS) is one of the most widely used multi-body dynamics simulation tools. It offers capabilities for modeling rigid and flexible bodies, applying various joint types, simulating nonlinear contacts, and performing time-domain dynamic analyses. ADAMS can be coupled with MATLAB/Simulink for co-simulation, making it valuable for control system development and vehicle dynamics applications.
- *Simpack (Dassault Systèmes):* Simpack is a high-fidelity simulation tool designed to model complex mechanical systems, particularly in the automotive, railway, and aerospace sectors. It supports real-time simulations, flexible body dynamics, and integration with finite element tools. Simpack is noted for its robustness in handling large nonlinear systems and supports both frequency-domain and time-domain analyses.
- *RecurDyn (FunctionBay Inc.):* RecurDyn combines multi-body dynamics with flexible body analysis using the finite element method (FEM). It features an intuitive graphical user interface, strong contact-modeling capabilities, and customizable automation scripts. RecurDyn also supports multiphysics co-simulation, enabling interaction with fluid dynamics and control systems.
- *Altair MotionSolve:* MotionSolve is part of the Altair HyperWorks suite and offers advanced simulation capabilities for both linear and nonlinear multi-body systems. It supports parametric optimization, flexible body modeling, and integration with other Computer-Aided Engineering (CAE) tools. MotionSolve is particularly used in virtual prototyping and design verification.

Open Source and Academic Tools

Open-source tools are gaining popularity in academia and research owing to their flexibility, transparency, and cost-effectiveness. These platforms allow for customization and integration with in-house codes and are often suitable for educational purposes or novel research applications.

- *Project Chrono:* Project Chrono is an open source, physics-based simulation framework that supports multi-body dynamics, collision detection, and real-time simulation. It offers Graphics Processing Unit (GPU) acceleration for large-scale systems and includes modules for granular flows, soft-body dynamics, and sensor simulation. The chrono is extensible and supports integration with the Robot Operating System (ROS), making it ideal for robotics and autonomous system research.
- *MBsysC (LIRMM, France):* MBsysC is part of the MBsysLab ecosystem and is designed for the modeling and simulation of multi-body systems with a focus on academic and research applications. It features modular architecture, symbolic pre-processing, and real-time control system integration. MBsysC supports holonomic and nonholonomic constraints, making it suitable for vehicle dynamics and robotic systems.
- *Modelica/Dymola:* Modelica is a non-proprietary, object-oriented modeling language used for multidomain system simulation. Dymola (Dynamic Modeling Laboratory) is a commercial

implementation that supports mechanical, electrical, thermal, and control system co-simulations. The use of equation-based modeling allows for compact representations and automatic generation of governing equations, making it particularly suitable for complex mechatronic systems.
- *Open Modelica:* Open Modelica is an open-source version of the Modelica environment that supports multidomain modeling and simulation. Although it is less feature-rich than Dymola, it is widely used in academia and supports system-level modeling with mechanical, thermal, and control elements.

These open-source tools, although sometimes limited in graphic capabilities compared to commercial platforms, provide valuable flexibility for developing new modeling methodologies and conducting fundamental research in multi-body system dynamics.

EMERGING TRENDS IN MBS SIMULATION

With the increasing complexity and integration of modern engineering systems, traditional approaches to MBS modeling are evolving rapidly. Advances in computational methods, data science, and sensor technologies have led to new paradigms for dynamic simulations. These trends aim to improve model fidelity, enhance predictive capabilities, enable real-time interactions, and facilitate cross-domain integration. This section presents four key emerging areas: flexible multi-body dynamics with finite element analysis (FEA), AI-driven modeling, digital twins, and Multiphysics-multiscale integration.

Flexible Multi-Body Dynamics with FEA

While traditional MBS modeling treats system components as rigid bodies, many engineering applications require accounting for structural flexibility, particularly in lightweight or high-speed systems. The integration of flexible body dynamics with FEA enables a more realistic simulation of the components that undergo elastic deformation during operation.

Flexible multi-body dynamics (FMBD) typically employs methods such as modal reduction or component mode synthesis (CMS) to reduce the complexity of finite element models while preserving essential dynamic characteristics. This approach is particularly relevant in aerospace, automotive, and robotics applications, where vibration behavior, stress distribution, and dynamic coupling need to be evaluated simultaneously.

AI-Driven and Data-Driven Modeling

Artificial intelligence (AI) and data-driven modeling techniques are transforming the manner in which mechanical systems are simulated and optimized. By leveraging machine learning algorithms trained on historical simulations or sensor data, engineers can develop surrogate models that approximate system dynamics with high speed and reasonable accuracy.

Applications of AI in MBS include system identification, fault detection, parameter tuning, and predictive maintenance. For instance, neural networks and Gaussian process models can be used to emulate the behavior of computationally expensive subsystems, allowing for faster simulations for control design or optimization studies.

Digital Twins and Real-Time Feedback

The concept of the digital twin—a virtual replica of a physical system that operates in parallel with real-time data—is revolutionizing system monitoring, diagnostics, and control. In the context of MBS, digital twins allow continuous simulation and state estimation of mechanical systems during actual operation.

Digital twins use sensor data to update MBS models in real-time, enabling predictive diagnostics, anomaly detection, and performance optimization. They are especially valuable in automotive, aerospace, and manufacturing industries, where downtime and maintenance costs are critical.

Multiphysics and Multiscale Integration

Modern engineering challenges increasingly involve the interaction of multiple physical domains: mechanical, thermal, fluid, electromagnetic, and chemical. Multiphysics integration in MBS simulations enables the study of such coupled phenomena, which are essential in applications such as thermal fatigue in actuators, aeroelastic flutter in aircraft, and electromechanical interactions in mechatronic systems.

Simulation platforms now support coupling between MBS solvers and computational fluid dynamics (CFD), electromagnetics, or thermodynamics modules. This level of integration is essential for designing high-performance systems, such as hybrid powertrains, MEMS devices, and advanced prosthetics.

RESEARCH CHALLENGES AND FUTURE DIRECTIONS

Despite significant progress in the modeling, simulation, and application of MBS, several research challenges persist. These limitations span computational, methodological, and integration domains, particularly as engineering systems grow in complexity and become more interconnected with control, sensing, and AI. Addressing these challenges is crucial for further advancing the fidelity, efficiency, and real-world applicability of MBS.

Handling High-Dimensional, Nonlinear Systems

One of the core challenges lies in the accurate simulation of large-scale systems with many degrees of freedom, nonlinear constraints, and coupled dynamics. As the number of components and interactions increases, so does the size and stiffness of the system's governing equations. Solving these equations efficiently without sacrificing stability or accuracy remains a critical issue, particularly in real-time or embedded applications. Advanced numerical solvers, parallel processing techniques, and model-order reduction strategies are essential research directions in this field.

Robust Modeling of Contact and Impact Events

Many practical systems involve intermittent contacts, such as gear meshing, foot-ground interactions in legged robots, and tire-road contact in vehicles. The accurate modeling of these events requires specialized algorithms to handle discontinuities, friction models, and deformation behavior. However, existing models often struggle with numerical stability or require fine discretization, which is computationally expensive. Improved hybrid modeling approaches that combine rigid and compliant contact representations along with efficient event-detection schemes are necessary for robust simulations.

Integration with Data-Driven and AI-Based Methods

While data-driven modeling and AI techniques are gaining traction, challenges remain in seamlessly integrating these methods with physics-based MBS frameworks. Issues such as data scarcity, generalization to unseen conditions, and a lack of interpretability hinder broader adoption. Future research should focus on hybrid modeling approaches that combine physical laws with machine learning to maintain accuracy while reducing computational costs. Additionally, explainable AI and physics-informed neural networks (PINNs) are promising directions that need further exploration in the context of MBSs.

Real-Time and Embedded Simulation Capabilities

The demand for real-time simulations is growing in areas such as digital twins, hardware-in-the-loop testing, and autonomous system control. Traditional high-fidelity MBS models often exceed real-time computation limits, particularly when flexible bodies or multiphysics coupling is involved. Research is needed to develop fast surrogate models, adaptive time integration schemes, and embedded solvers optimized for edge-computing platforms. Efficient data exchange protocols between physical systems and their digital counterparts are also essential for enabling continuous synchronization.

CONCLUSION

The dynamic modeling and simulation of MBS represent a cornerstone of modern mechanical engineering, enabling comprehensive analysis of complex, interconnected mechanical components across diverse domains. This review examined foundational modeling approaches—Newton–Euler, Lagrangian, and Kane’s formulations—alongside critical techniques for constraint stabilization and numerical integration. These methods form the analytical backbone for developing accurate and scalable models capable of capturing real-world system behavior under dynamic operating conditions.

Advancements in simulation software, both commercial and open source, have democratized access to high-fidelity MBS modeling tools, allowing researchers and engineers to visualize, simulate, and optimize system performance across application domains, such as automotive engineering, robotics, aerospace mechanisms, and biomechanics. The integration of flexible body dynamics through finite element coupling further enhances simulation realism, particularly in high-speed or lightweight systems, where structural deformation significantly influences system behavior.

Emerging trends, such as data-driven modeling, AI integration, real-time digital twins, and multiphysics coupling, are reshaping the landscape of MBS simulations. These innovations not only extend the functionality and efficiency of simulation platforms but also pave the way for intelligent, adaptive, and predictive mechanical systems. As systems become more autonomous, interconnected, and performance-critical, the ability to model and simulate them in real time with high fidelity is becoming increasingly vital.

REFERENCES

1. Shabana AA. Dynamics of Multibody Systems. 4th ed. Cambridge: Cambridge University Press; 2013. doi:10.1017/CBO9781107337213.
2. Haug EJ. Computer-Aided Kinematics and Dynamics of Mechanical Systems. Boston: Allyn & Bacon; 1989.
3. Kane TR, Levinson DA. Dynamics: Theory and Applications. New York: McGraw-Hill; 1985.
4. Pfeiffer F, Glocker C. Multibody Dynamics with Unilateral Contacts. Chichester: John Wiley & Sons; 1996. doi:10.1002/9783527618385.
5. Nikravesh PE. Computer-Aided Analysis of Mechanical Systems. Englewood Cliffs: Prentice Hall; 1988.
6. Tasora A, Negrut D. Project Chrono: an open-source framework for the physics-based simulation of rigid and soft body dynamics. Multibody Syst Dyn. 2021;52(2):139–72.
7. Pettersson L, Persson LE. Flexible body modeling in multibody systems using component mode synthesis. Multibody Syst Dyn. 2002;8:317–34.
8. García de Jalón J, Bayo E. Kinematic and Dynamic Simulation of Multibody Systems: the Real-Time Challenge. New York: Springer-Verlag; 1994. doi:10.1007/978-1-4612-2600-0.
9. Sinha A, Rakheja S, Sankar T. A review of dynamic modeling of mechanical systems with clearances. Mech Mach Theory. 1997;32(6):705–24.
10. Altair Engineering, Inc. MotionSolve User Guide. Troy (MI): Altair Engineering; 2022.
11. Modelica Association. Modelica Language Specification. Version 3.6. Modelica Association; 2023.
12. Negrut D, Serban R, Melanz D, Tasora A. High-performance computing techniques for modeling and simulation of ground vehicle mobility systems. Comput Methods Appl Mech Eng. 2020;360:112697.