



AN OVERVIEW OF FORWARD DYNAMICS ALGORITHMS AND THEIR USE IN OPEN-SOURCE DYNAMICS ENGINES

Nikola L.J. Zivkovic¹, Jelena Z. Vidakovic² and Mihailo P. Lazarevic³

¹Lola Institute, 11030 Belgrade, Serbia, e-mail: nikola.zivkovic@li.rs

¹Lola Institute, 11030 Belgrade, Serbia, e-mail: jelena.vidakovic@li.rs

²Faculty of Mechanical Engineering, University of Belgrade, Belgrade, Serbia, e-mail: mlazarevic@mas.bg.ac.rs

Keywords: Forward dynamics, Robotics, Dynamics engine, Open-source, Simulator.

ABSTRACT

Simulation of real-world dynamics is of major importance in testing and verifying developed industrial concepts and solutions, developing and verifying potential control paradigms, scientific research, learning and training tools, or the entertainment industry as a basis for a game engine. The module of the 3D virtual simulator that achieves simulation of the real-world behaviour such as rigid and elastic body dynamics, particle dynamics, fluid dynamics, electrodynamics, magnetism, etc., is often referred to as a dynamics engine or physics engine. The core of the rigid body dynamics (physics) engine is the solution to the forward dynamics problem, which is defined as finding a rigid body's path, velocity, and acceleration for a given input actuating torque and external forces. The past few decades saw a considerable amount of research in robot dynamics modelling, and there are many methods for robot dynamic model development available in the literature. The most commonly used algorithms for solving robot forward dynamics problem are the Composite-Rigid-Body Algorithm (CRBA) [1] and the Articulated-Body Algorithm (ABA) [2]. CRBA and ABA are reduced coordinate methods where known constraints, such as joints, are embedded in the formulation of the equations of motion. Besides reduced coordinate methods, there are maximal coordinate methods using Lagrange multipliers [3-4] to enforce constraints using constraint reaction forces.

This paper presents a comprehensive overview of forward dynamics algorithms and their usage in dynamics engines. Special reference is given to the most commonly used algorithms and methods and their advantages and disadvantages depending on the application. Most important software intended for virtual simulation of robots is presented, emphasising free, open-source use.

Firstly, brief history and introduction of CRBA, ABA and Lagrange multipliers methods is given, as they are the most commonly used methods employed by dynamics engines. Next, general phases of the simulation process are described. An integral segment in creating a simulation is the definition of the world - a description of the environment and robot models that are to be simulated. Application of the actuation and external forces and torques to the model, detection of collisions between the bodies, constraint solving, forward dynamics computation and integration to obtain velocity and position of the bodies are performed. Each of these aspects is described with special attention to constraint solving and computation of forward dynamics using the algorithms mentioned above. There is a myriad of free and open-source dynamics engines available, and the focus herein is on the most commonly used engines for simulating robots: Open Dynamics Engine (ODE) [5], DART (Dynamic Animation and Robotics Toolkit) [6], Bullet [7], and Simbody [8] as they are present in the two most popular free, open-source robotics simulators Cyberbotics Webots [9] and OpenRobotics Gazebo [10]

(Ignition is the successor of Gazebo).

It can be concluded that the most used reduced coordinate method in simulators is ABA, while Lagrange multipliers are the most popular maximal coordinate method. ABA is mainly used for simulating open-loop multi-body systems (robot manipulators) where joint constraints are known upfront. In contrast, Lagrange multiplier methods are used where modularity of the simulated model during simulation run-time is crucial. In reality, most dynamic engines have access to both of these methods to ensure the diversity of the simulation processes that can be executed.

Presented conclusions are useful for the appropriate selection of available simulation methods depending on the application, as well as within further advancement and development of simulation and verification frameworks for robotic manipulators and rigid body systems.

REFERENCES

- [1] Walker M. W., Orin D. E. (1982), "Efficient Dynamic Computer Simulation of Robotic Mechanisms," *J. Dyn. Sys., Meas., Control.*, vol. 104, no. 3, pp. 205-211, Sept., doi: 10.1115/1.3139699.
- [2] Featherstone R. (2008), "Forward Dynamics - Propagation Methods," in *Rigid Body Dynamics Algorithms*, Boston, MA, U.S.: Springer, ch. 7, pp. 119-140.
- [3] Baraff D. (1996), "Linear-Time Dynamics using Lagrange Multipliers," *SIGGRAPH*, New Orleans, Louisiana, U.S, pp. 137-146.
- [4] Hernandez F., Garre C., Casillas R., Otaduy M. A. (2011), "Linear-Time Dynamics of Characters with Stiff Joints", *Proc. of V Ibero-American Symposium in Computers Graphics - SIACG*, pp. 163-168.
- [5] Open Dynamics Engine. (2001-2004), Russell L. Smith. [Online]. Available: <https://www.ode.org/>
- [6] DART (Dynamic Animation and Robotics Toolkit). (2021), Georgia Tech and Carnegie Mellon University. [Online]. Available: <https://dartsim.github.io/index.html>
- [7] Bullet. (3.24). [Online]. Available: <https://pybullet.org/wordpress/>
- [8] Simbody: Multibody dynamics API. (3.7), SimTK. [Online]. Available: <https://simtk.org/projects/simbody/>
- [9] Webots. (2022), Cyberbotics. [Online]. Available: <https://cyberbotics.com/>
- [10] Gazebo. (2022), Open Robotics. [Online]. Available: <https://gazebosim.org/home>