Proceedings of the 2018 Winter Simulation Conference
M. Rabe, A.A. Juan, N. Mustafee, A. Skoogh, S. Jain, and B. Johansson, eds.

PHYSX AS A MIDDLEWARE FOR DYNAMIC SIMULATIONS IN THE CONTAINER LOADING PROBLEM

Juan C. Martínez-Franco

Department of Mechanical Engineering
Universidad de Los Andes
Carrera 1 Este No. 19A – 40
Bogotá, 11711, COLOMBIA

David Álvarez-Martínez

Department of Industrial Engineering
Universidad de Los Andes
Carrera 1 Este No. 19A – 40
Bogotá, 11711, COLOMBIA

ABSTRACT

The Container Loading Problem (CLP) is an optimization challenge where the constraint of dynamic stability plays a significant role. The evaluation of dynamic stability requires the use of dynamic simulations that are carried out either with dedicated simulation software that produces very small errors at the expense of simulation speed, or real-time physics engines that complete simulations in a very short time at the cost of repeatability. One such engine, PhysX, is evaluated to determine the feasibility of its integration with the open source application PackageCargo. A simulation tool based on PhysX is proposed and compared with the dynamic simulation environment of Autodesk Inventor to verify its reliability. The simulation tool presents a dynamically accurate representation of the physical phenomena experienced by cargo during transportation, making it a viable option for the evaluation of dynamic stability in solutions to the CLP.

1 INTRODUCTION

The Container Loading Problem (CLP) consists of the geometric arrangement of small rectangular items (cargo) into a larger rectangular space (container), but it is not a simple geometry problem. The arrangements of cargo must maximize volume utilization while complying with certain constraints such as cargo fragility, delivery order, etc. The CLP has been proven to be NP-Hard (Pisinger 2002), which in addition to its large spectrum of applicability, has kept researchers and industry alike interested in the problem. In recent years, new constraints for the CLP have become intertwined with the physical behavior of cargo.

Real CLP applications require the modelling and consideration of practical constraints of significant complexity. Bortfeldt and Wascher (2013) present a compilation of constraints related to the container loading problem. Among several types of constraints, this work focuses on cargo stability. The cargo stability constraint seeks to preserve the integrity of the items and the safety of operators carrying out loading and unloading operations. Previous works that address cargo stability on the CLP have defined two types of stability: static and dynamic. Static stability refers to the static equilibrium of the cargo during loading and unloading, whereas dynamic stability refers to the equilibrium of the cargo during transportation, and is rarely addressed in literature.

Ramos et al. (2015) proposed a set of dynamic stability metrics for the CLP, which reflect dynamic stability based on the effects of movement on a cargo arrangement, while Ramos et al. (2014) introduced the first simulation tool used to study the phenomenon of dynamic stability in the CLP. As this tool is closed source, it cannot be modified to solve more realistic variants of the CLP than the ones classically considered. With this in mind, this work presents a novel, open source tool to carry out dynamic simulations that provide aid in evaluating the dynamic stability of solutions to the CLP. This tool is expected to be used to advance existing solutions and to find new ones for real-world problems in the logistics industry.

Dynamic simulations consist of the use of a computer program to run a mathematical model detailing the behavior of a physical system, usually described as sets of differential equations. As these sets of equations typically grow in complexity to become non-linear, algorithms based on numerical methods become necessary to solve them. Programs meant exclusively for this purpose are called high precision simulation software, while middleware applications meant to integrate dynamic simulations into other applications are called Physics Engines.

Physics engines range in accuracy from high precision, deterministic software used in aircraft design and stress analysis tools to stochastic middleware intended for real-time simulation in interactive applications. Regarding the study of dynamic behavior in the CLP, a physics engine specialized in rigid body dynamics and Newtonian mechanics is of most interest. These two concerns are usually the least computationally expensive physical systems to simulate, and as such, even real-time physics engines should run them with acceptable accuracy (Boeing and Bräunl 2007). Nevertheless, this article presents an evaluation of PhysX (NVIDIA 2018) as a viable platform for running dynamic simulations in the CLP.

Initially, physics considerations for a sufficiently accurate model of packing patterns will be presented, which will then be associated with necessary features in physics engines. Subsequently, PhysX will be evaluated to verify its theoretical compliance with said features, after which a series of simulations will be performed on both PhysX and a suite of high precision simulation software, Autodesk Inventor, to determine the reliability of the former. Finally, a simulation tool making use of PhysX will be presented and used to conduct a series of tests related to dynamic stability in the container loading problem.

The paper is organized as follows. Section 2 includes the physics considerations in the CLP. Section 3 introduces the methodology used for the creation and validation of the simulation tool. Results are presented in Section 4, followed by conclusions in Section 5.

2 PHYSICS CONSIDERATIONS

2.1 The Container Loading Problem

Solutions to the CLP take the form of packing patterns that define the spatial configurations of boxes or pallets within a container. Dynamic stability constraints require the cargo to maintain its integrity when transported. The Container Handbook (GDV 2018) presents the most common accelerations, besides gravity, that occur in the transportation of cargo on land roads as defined by different regulations; these accelerations are summarized in Table 1.

As the least conservative accelerations, and the ones that influence the dynamic behavior of cargo the most, the British regulations will be used as a reference for the simulation parameters, but the option to change the magnitude of the accelerations will be included.

2.2 Assumptions

A packing pattern is for most intents and purposes modelled as a collection of smaller rectangular cuboids representing boxes or pallets and a larger cuboid representing the container. In simulation, physical assumptions commonly applied are:

- **Constant density.** This means that the mass is equally distributed for the box, the geometric center corresponding to the center of gravity.
- **Uniform gravity field.** Acceleration due to gravity is downward, and its magnitude (9.81 m/s^2) is constant for every point in the work space. A classic assumption, very close to the truth in practical scenarios. Only discarded when dealing with extremely large quantities of mass or vast spaces.

- **Invariable friction parameters.** Often a single coefficient of static friction and a single coefficient of dynamic friction are considered, with the same friction model applying for every contact surface and friction only influenced by normal forces on said surfaces.
- **Invariable coefficient of restitution.** All collisions are to be solved using the same coefficient of restitution. Typically, a coefficient is chosen such that all collisions are inelastic, meaning that the boxes will not "bounce" on impact.
- **Rigid bodies.** Neither the boxes nor the containers are assumed to deform, regardless of the loads applied to them.

Table 1: Accelerations for road transport found in the Container Handbook.

Norm	Forward acceleration	Braking acceleration	Lateral acceleration
Verein Deutscher Ingenieure	0.8g	0.5g	0.5g
International Maritime Organization	1.0g	0.5g	0.5g
Swiss regulations	1.0g	0.5g	0.5g
British regulations	1.2g	0.5g	0.8g

2.3 Validity of the Model

In a real-world cargo configuration, these assumptions are not the closest approximations of reality; the sides of boxes or restricted pallets are never perfect parallelepipeds due to practical manufacturing tolerances and are deformable to a significant degree, also presenting unmodeled irregularities that result from wrappings. The center of gravity rarely corresponds to the geometric center, as packages are often unbalanced due to the differences in density between transported goods and protective material. Therefore, it is difficult to accurately predict the behavior of stacked objects. Further difficulties arise from small variations in materials that affect friction models, as well as a plethora of other effects that alter the way force is transmitted between stacked objects. However, for objects of sizes and masses within the orders of magnitude of typical cargo, the acting forces and reactions are large enough that, when considered in bulk, inaccuracies smooth out over several instances.

3 METHODOLOGY

3.1 Necessary Engine Features

Given the physical considerations presented in Section 2, it was determined that the physics engine to use should be performant at collision detection and rigid body dynamics, while soft body dynamics are not relevant, and fluid dynamics are of little concern, with air drag being negligible at the velocities expected to occur inside a container during transportation.

However, a particular problem arises from the simulation of forces acting upon packing patterns. When boxes are not tethered but rather piled up on top of each other, traditional solvers for friction that use a simplified Coulomb model rarely produce repeatable results in physical experiments (Boeing and Bräunl 2007). A traditional approach consists of introducing noise into the solver to account for irregularities that affect the transmission of force through stacked objects. Non-deterministic physics engines where float imprecision plays a part have also been observed to simulate stacked objects in a verisimilar, if not a mathematically exact, manner (Boeing and Bräunl 2007).

3.2 PhysX

PhysX is a physics engine frequently used in the real-time simulation of physics and is frequently implemented in videogames. This means that the engine is capable of trading accuracy for simulation speed by forcing calculations to stop if the time consumed exceeds a maximum value. In other words, the integrator uses as fixed timestep. This differentiates PhysX from most scientific-targeted simulation software, as the truncation of calculations due to variations in execution speed can lead it to be non-deterministic. This is problematic when information such as the dynamic stability indicators is to be obtained from the simulation to incorporate into the optimization process of the CLP. However, as explained above, this characteristic is also related to the solution of the stacked objects problem without the need to introduce noise to the simulation. Other limitations, such as lack of soft body simulation and Coriolis accelerations, are not relevant when simulating to obtain dynamic stability metrics.

3.3 Comparison with Autodesk Inventor

Autodesk Inventor is a parametric modelling program frequently used in computer assisted mechanical design. It contains a dynamic simulation environment with a solver based on Runge-Kutta integration capable of evaluating a mechanical system's behavior when affected by conditions of force, torque, mechanical joints, and imposed motion (Autodesk 2018). Inventor's friction model is based on the definition of a single friction coefficient that modifies a contact force function which has among its parameters the normal force and the relative velocity of the contact surfaces. This contrasts with PhysX's simpler kinetic and dynamic friction separation.

The simulation environment implements a variable timestep, where calculations will continue until the integration error is sufficiently small. This means that the simulation results can be as precise as necessary. Additionally, this software will always produce the same results on multiple runs of the same system as long as initial conditions and simulation parameters remain unchanged. However, dynamic simulations in Autodesk Inventor are computationally expensive. A simulation running four seconds of acceleration of a packing pattern inside a container will take several minutes to finalize. This makes the software ill-fitted to be integrated into optimization-heavy workflows, like the CLP.

Nevertheless, the deterministic nature of the software and adjustable error indicate that simulation results produced in Inventor serve as a reference benchmark for the accuracy of the results obtained through PhysX.

3.4 Simulation Tool (PackageCargo)

A simulation tool to evaluate the evolution of a packing pattern was developed in Unity, a development platform for videogames that implements the PhysX SDK as a backend for physics and the .NET framework as an API backend. The tool was integrated into an open source application, PackageCargo (Martínez-Franco et al. 2018a) which, in addition to performing dynamic simulations, can produce and visualize packing patterns to instances of the CLP considering some of the most relevant constraints published on the literature, due to its optimization module that uses the GRASP algorithm presented by Álvarez-Martínez et al. (2015). A screenshot of the simulation tool is shown in Figure 1. PackageCargo can also predict some dynamic stability metrics such as the number of fallen boxes by using a mechanical model embedded in its physics module, based on the algorithm presented by Martínez-Franco et al. (2018b).

The application includes a GUI to set some simulation parameters, like accelerations different to those defined by the British regulations, the time during which those accelerations are maintained, and the simulation time scale, an indirect way to define the length of the time step. Accelerations are to be applied in the form of sinusoidal functions, as seen in Figure 2 to prevent unwanted behaviors resulting from jerk.

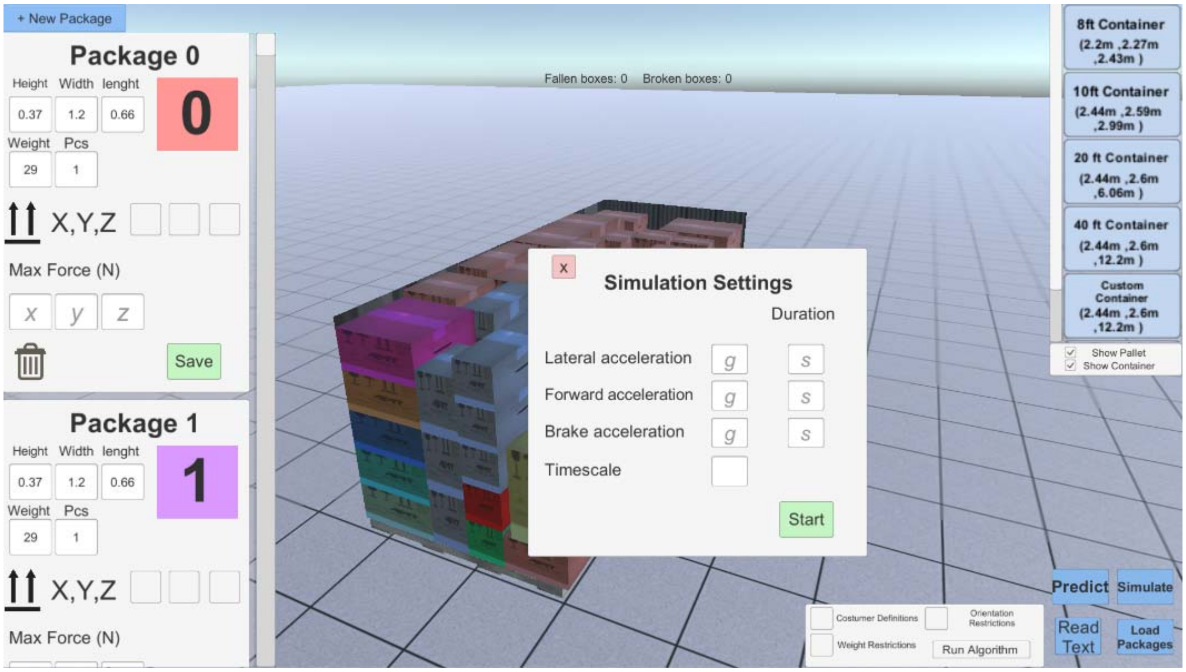

Figure 1: Screenshot of PackageCargo, including the simulation tool.

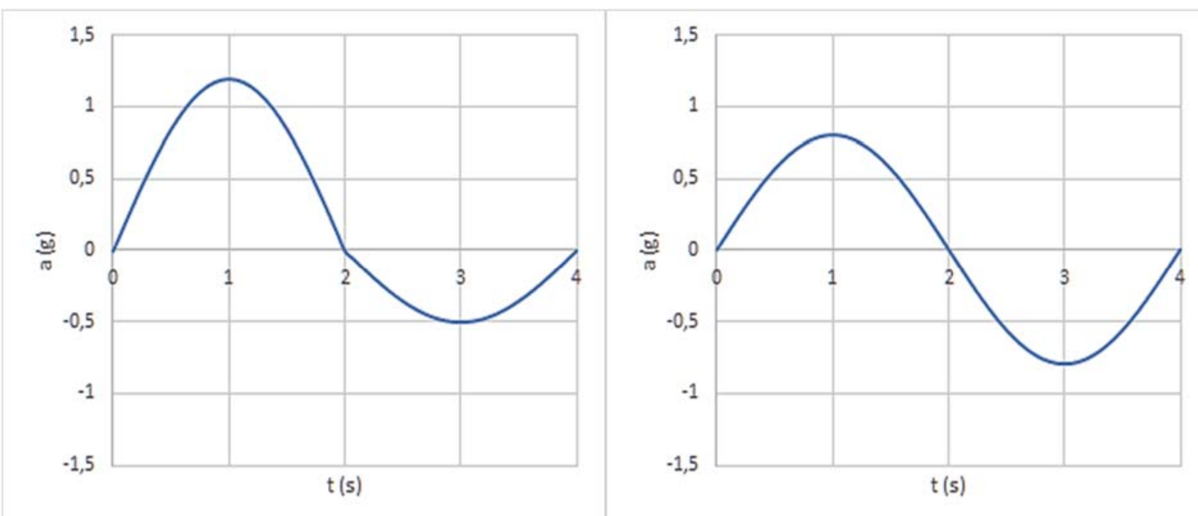

Figure 2: Acceleration functions in the forward direction (left) and the lateral direction (right).

4 RESULTS

All simulations were run on a 64-bit Windows 10 machine with an Intel Core i7-7700HQ processor, 16 GB of RAM and a 4GB NVIDIA GeForge GTX 1050 Ti GPU. The Inventor 2017 and Unity 5.6 versions where used.

4.1 PhysX Accuracy

A benchmark test was performed, consisting of stacking three identical boxes within both Inventor and Unity, and then applying an acceleration of 1 g (9.81 m/s^2) to the middle box (colored red on Figure 3), in a similar manner to a "tablecloth trick".

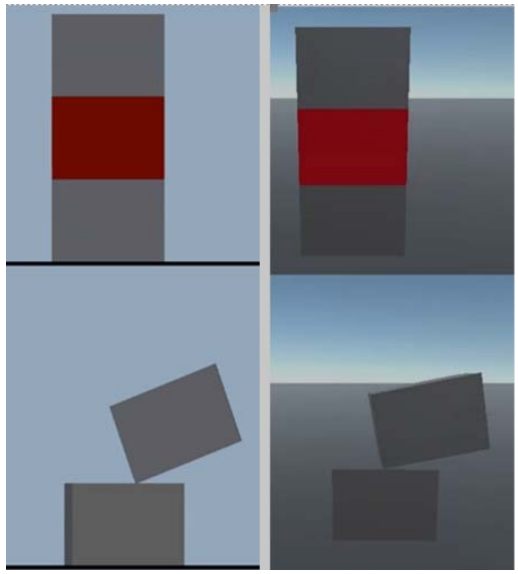

Figure 3: Benchmark test capture of initial and final states on Inventor (right) and Unity (left).

Results were visually compared and determined to be similar enough to indicate that the developed application has an adequate precision. Nevertheless, a second test was devised to compare velocity tracking in both simulation tools.

4.2 CLP Simulation

The most relevant results were obtained from simulating the behavior for a packing pattern in PackageCargo and replicating said pattern and external loads within Inventor. Figure 4 shows a three-dimensional representation of the packing pattern originating from PackageCargo and recreated within Autodesk inventor.

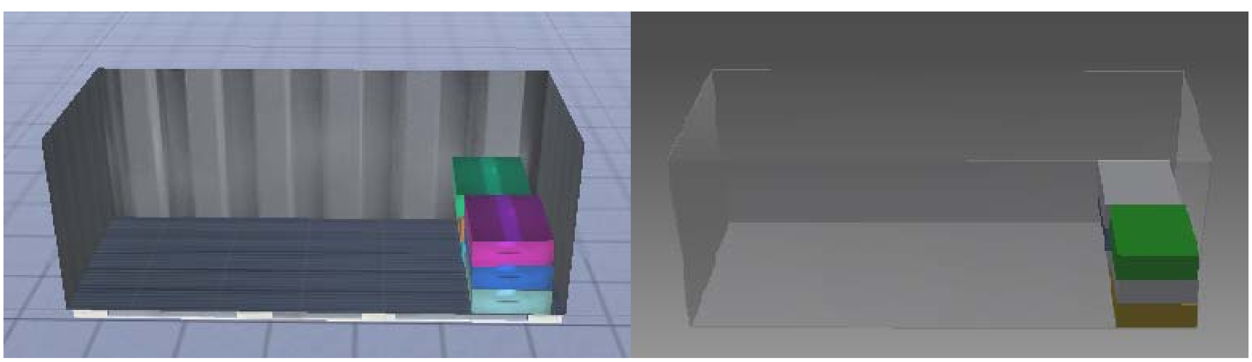

Figure 4: Dynamic simulation comparison in Unity (right) and Inventor (left).

To compare the physical behavior of the pattern in both tools, velocity was tracked for each of the six packages, with the final velocity components shown on Table 2. As instantaneous velocity is of great importance when evaluating dynamic stability. Therefore, these results provide important insight regarding the effectiveness of the simulation tool.

Table 2: Final velocity by axis for an 8 second simulation of a packing pattern in motion.

Box ID	Autodesk Inventor			PackageCargo		
	x Velocity	y Velocity	z Velocity	x Velocity	y Velocity	z Velocity
Box 1	-8.899	0.0305	-1.551	-8.908	0.0046	-1.594
Box 2	-8.901	-0.0236	-1.560	-8.923	0.0034	-1.595
Box 3	-8.905	0.0160	-1.570	-8.911	0.0019	-1.612
Box 4	-8.895	0.0014	-1.570	-8.921	0.0025	-1.601
Box 5	-8.896	0.0370	-1.591	-8.910	0.0081	-1.611
Box 6	-8.887	0.0232	-1.593	-8.910	0.0134	-1.607

Results are very similar, with the topmost boxes (5 and 6) presenting the lowest velocity values, which is to be expected as these boxes present the greatest tendency to slide. The simulation in PackageCargo produces little vertical velocity in the boxes, resulting from a simplified contact model in PhysX and the fact that collisions can never be completely inelastic in Inventor, as contact parameters must have non-zero rigidity and a finite dampening coefficient. Notably, it took close to 17 minutes to complete the simulation in Inventor, while the simulation in PackageCargo was run in real time, which is to say, in 8 seconds. This difference is expected to increase dramatically with more cargo items, as Inventor is inefficient when dealing with multiple 3D contact joints (Autodesk 2018), while PhysX has been optimized to deal with multiple colliding bodies (NVIDIA 2018).

5 CONCLUSIONS

The Container Loading Problem does not require simulation features beyond those that most physics engines are capable of, even considering difficulties when solving for stacked bodies. PhysX is accurate enough according to the performed benchmark test and the comparison simulations for a 6-box packing pattern. Moreover, PackageCargo completed the test simulation in seconds, whereas Inventor required several minutes to perform the same task. Future work might focus on experimental verification not only of the results of the simulations, but also of the assumptions considered for dynamic stability.

REFERENCES

Álvarez-Martínez, D., D. Alvarez-Valdes, and F. Parreño. 2015. "A GRASP Algorithm for the Container Loading Problem with Multi-drop Constraints". *Pesquisa Operacional* 35(1):1-24.

Autodesk. 2018. Autodesk Inventor Professional 3D CAD software. http://www.autodesk.com, accessed March 3rd, 2018.

Boeing, A., and T. Bräunl. 2007. "Evaluation of Real-time Physics Simulation Systems". In *Proceedings of the 5th International Conference on Computer Graphics and Interactive Techniques in Australia and Southeast Asia* – GRAPHITE '07, 281–288. New York, NY, USA: ACM Press.

Bortfeldt, A., and G. Wascher. 2013. "Constraints in Container Loading A State-of-the-art Review". *European Journal of Operational Research* 229(1):1–20.

GDV. 2018. Gesamtverband der Deutschen Versicherungswirtschaft. Container Handbook. Cargo Loss Prevention Information from German Marine Insurers. Chapter 2. http://www.containerhandbuch.de/chb_e/, accessed March 24th, 2018.

Martínez-Franco J. C., D. H. Cuellar, E. S. Caceres, and D. Álvarez-Martínez. 2018a. PackageCargo – Open Source Tool Based on Optimization and Simulation for the Container Loading Problem with Dynamic Stability, In *Proceedings of the International Conference on Industrial Engineering and Operations Management,* July 26th-27th, Paris, France.

Martínez-Franco J. C., D. H. Cuellar, and D. Álvarez-Martínez. 2018b. "Review of Dynamic Stability Metrics and a Mechanical Model Integrated with Open Source Tools for the Container Loading Problem", *Electronic Notes in Discrete Mathematics* 69C:325-332.

NVIDIA Corporation. 2018. PhysX. https://developer.nvidia.com/PhysX-sdk, accessed March 25th, 2018.

Pisinger, D. 2002. "Heuristics for the Container Loading Problem". *European Journal of Operational Research* 141(2):382-392.

Ramos, A., J. Oliveira, and A. M. Gomes. 2014. "A Physics Simulation Tool for the Container Loading Problem". In *Proceedings of the European Modeling and Simulation Symposium*, September 10th-12th, Bordeaux, France.

Ramos, A., J. Oliveira, J. Goncalves, and M. Lopes. 2015. "Dynamic Stability Metrics for the Container Loading Problem". *Transportation Research Part C: Emerging Technologies* 60: 480-497.

AUTHOR BIOGRAPHIES

JUAN C. MARTÍNEZ-FRANCO received both an undergraduate and a master's degree in Mechanical Engineering from Universidad de Los Andes, Bogotá, Colombia, in 2018. His e-mail address is jc.martinez10@uniandes.edu.co.

DAVID ÁLVAREZ-MARTÍNEZ received a PhD from Universidade Estadual Paulista "Júlio de Mesquita Filho" (Brazil) in 2014. He is a Professor of Industrial Engineering at Universidad de Los Andes (Colombia) since 2016. His current research interest includes optimization, simulation and industrial automation (robotics) for transportation and logistics problems. His e-mail address is d.alvarezm@uniandes.edu.co.