

Two-Dimensional Modeling of Explosion Impulses on a Structure Using the Discrete Element Method

Martyniuk P. M., Kochkarov V. D.

*Department of Computer Science and Applied Mathematics,
National University of Water and Environmental Engineering,
11 Soborna str., 79000, Rivne, Ukraine*

(Received 1 January 2025; Revised 29 October 2025; Accepted 1 November 2025)

Explosive phenomena pose significant threats to structural integrity, necessitating accurate prediction models for designing resilient infrastructure. Traditional computational approaches like CFD and FEM, while detailed, require substantial computational resources and specialized expertise. This study presents an alternative approach using the Discrete Element Method (DEM) implemented through the Pymunk physics engine for 2D explosion modeling. The developed method models explosions as radially distributed particles with initial impulses, simulating shock wave propagation through particle collisions. Structures are represented using a modular approach, enabling detailed analysis of impulse distribution across different building elements. The simulation tracks collision events and calculates impulse transfer using momentum conservation principles. Model validation was performed against UFC 3–340–02 standards by investigating three scaling methods: proportional coefficient, linear regression, and a non-linear power-law model. The power-law model demonstrated the best agreement with reference data, confirming the model's accuracy with a total integral error of only 1.5%. This computationally efficient approach provides a practical tool for structural engineers and urban planners to incorporate blast resistance considerations without requiring high-performance computing resources. The method successfully balances computational efficiency with physical fidelity, making explosion modeling more accessible for rapid assessment scenarios and preliminary design stages.

Keywords: *explosion; blast wave; impulse; physical modeling; Discrete Element Method.*

2010 MSC: 74A15, 74F05, 74M15, 65Z05

DOI: 10.23939/mmc2025.04.1157

1. Introduction

Explosive phenomena pose significant threats to structural integrity and public safety. The ability to accurately predict and model the consequences of explosions for structures is critically important for designing resilient buildings, planning urban infrastructure, and implementing effective protective measures. Despite a significant amount of research, explosion dynamics remains challenging to model due to its nonlinear nature and the multitude of factors influencing shock wave propagation and structural response.

The complexity of modeling explosions has led to the development of various computational approaches, each with its own advantages and limitations. Computational Fluid Dynamics (CFD) provides a detailed description of shock wave behavior but requires significant computational resources and careful mesh configuration. Recent comparative studies [1] have revealed substantial discrepancies in predicted peak pressure values and impulse among different CFD implementations, attributed to discretization levels, post-detonation process calculation methods, and boundary condition settings. Software implementations such as ANSYS Autodyn, blastFoam, ProSAir, and Viper::Blast demonstrate these variations, especially in far-field explosion scenarios.

The Finite Element Method (FEM), while effective for evaluating structural deformations and failures, often requires integration with gas dynamics calculations for comprehensive modeling of explosion effects. Such interaction adds complexity to the simulation process and increases computational re-

quirements. The Smoothed Particle Hydrodynamics (SPH) method offers an alternative approach based on particle interactions without using a mesh, which proves useful for scenarios with extreme deformations and complex explosive charge geometries, as demonstrated in reference [2]. SPH can function independently or in combination with Eulerian/Lagrangian schemes, although it has drawbacks in the form of increased resource consumption with an increasing number of particles and requires a more complex procedure for calibrating equations of state.

Empirical and semi-empirical methods, represented by tools such as CONWEP and EMBlast, provide rapid initial estimates of peak pressure and impulse based on established experimental databases. Although these approaches do not account for complex reflections, diffraction, and interaction effects with realistic geometries, they remain valuable for initial design stages and operational assessments [1]. In recent years, active comparisons of specialized explosion research software have been conducted, with detailed analysis of Autodyn, blastFoam, ProSAir, and Viper::Blast, revealing significant differences in calculation speed and impulse reproduction accuracy.

In this study, we utilized simulation modeling, specifically the Discrete Element Method (DEM), as an alternative to the aforementioned methods. The DEM approach models explosion wave dynamics through particle collisions, providing a physically intuitive representation of explosion propagation. In this research, a simplified but physically correct 2D simulation system was developed for modeling the impact of explosions on structures. Using the Pymunk physics engine, a mechanism was implemented where explosions are presented as radially distributed particles with initial impulses, enabling quick and sufficiently accurate reproduction of shock wave propagation.

This approach, based on the discrete element method and verified using data from the Unified Facilities Criteria document [3], maintains a balance between high computational efficiency and achievable level of physical fidelity, making explosion modeling more accessible for applied engineering tasks. By reducing computational complexity while preserving explosion physics principles, our method fills a gap in accessible explosion modeling, especially for scenarios requiring rapid assessment. The resulting framework provides a practical tool for structural engineers and urban planners to incorporate blast resistance considerations into their designs without requiring access to specialized high-performance computing resources or extensive training in complex modeling software.

2. Materials and methods

2.1. Main characteristics of explosive waves

An explosion is characterized by the rapid release of energy in the form of heat, light, sound, and shock wave. By definition, an explosion is a sudden conversion of potential energy into kinetic energy with the formation and release of gas under pressure [4]. This sudden release of energy creates a volume of highly compressed gas that rapidly expands, displacing the surrounding environment and forming a shock wave. During an explosion, a shock wave is formed, characterized by a sharp pressure jump followed by exponential decay, with the wave propagating at a speed exceeding the speed of sound in air [5].

The main parameters that determine the impact of an explosion on structures are the peak overpressure value, its duration, and impulse.

The dynamics of pressure change over time are described by various equations, which are essentially mathematical models of the physical process. The most well-known among them is the Friedlander equation [4], which is widely used for modeling the explosive wave profile [6], as presented in equation 1,

$$p(t) = p_0 + p_2 \left(1 - \frac{t}{t_2} \right) e^{-b \frac{t}{t_2}}, \quad (1)$$

where $p(t)$ is pressure as a function of time, p_0 is atmospheric pressure, p_2 is peak overpressure, t is time from the explosion onset, t_2 is the duration of the positive phase, b is the wave parameter. A detailed mathematical description of these dependencies and the interaction of the explosive wave with structures is presented in the work of Lomazzi, Giglio, & Manes (2021) [6], where the authors

investigate the phenomenon of interaction between the explosive wave and structure during explosive reactions and analyze energy transfer mechanisms that are critical for the development of protective measures. In the study by Ullah A., Ahmad F., Jang H. W., Kim S. W., & Hong J. W. [7], the authors conduct a comprehensive review of analytical and empirical methods for estimating explosion pressure. The authors analyze and compare various equations for calculating explosive wave profiles, focusing on unconfined spherical and hemispherical explosions from solid chemical explosives.

Relationship between impulse and pressure in explosions. For studying the complex phenomenon of explosions, it is important to understand the relationship between pressure and impulse, as impulse is an integral component of pressure. According to the “pressure–impulse” theory [8], impulse is defined as the integral of pressure over time:

$$I = \int_{t_0}^{t_1} p(t) dt, \quad (2)$$

where $p(t)$ is pressure as a function of time, and I is the pressure impulse.

The impulse represents the amount of momentum transferred by the explosive wave to an object in its path over a given time interval. For structural response assessment, both positive and negative phase impulses of the shock wave are analyzed. In this work, we currently consider only the impulse of the positive phase of the shock wave.

In our 2D simulation approach, the main focus is on the direct transfer of impulse between explosive particles and structural elements. By modeling an explosion as a set of particles with radial initial velocities, we can effectively reproduce the propagation of the shock wave and its interaction with buildings. This approach allows us to consider the main physical mechanisms of impulse transfer with minimal computational costs. Additionally, this information will serve as a foundation for further model expansion, such as calculating peak overpressure.

2.2. Discrete Element Method (DEM)

In our study, we used the Discrete Element Method (DEM), implemented through the Pymunk physics engine, which is based on the Chipmunk library [9]. The main reasons for choosing this approach include:

- **Computational efficiency:** The DEM method provides a balance between accuracy and computational speed, allowing for the simulation of various scenarios even with limited resources.
- **Simplicity and clarity:** Using particles for modeling explosions allows for visualizing physical processes, making the model accessible to researchers without specialized training.
- **Modularity:** The method allows for easily the modeling of structures from individual elements, allowing for consideration of local impulsive loads.
- **Direct modeling of impulse transfer:** The approach directly considers the transfer of impulse through particle collisions, which is crucial for evaluating explosive effects.

Despite the fact that this approach is inferior to the above-mentioned methods in terms of physical process detail, it is well-suited for analyzing impulsive loads on structures. The simplicity of implementation and the ability to conduct numerous experiments make it suitable for preliminary assessments and scientific goals.

Traditional empirical and analytical methods for blast pressure estimation, while computationally efficient, suffer from significant limitations in their applicability to complex scenarios and geometric configurations [7]. These approaches typically rely on simplified assumptions about charge geometry, free-field conditions, and uniform material properties that may not adequately represent real-world blast scenarios. The DEM approach addresses these limitations by providing a physics-based framework that can accommodate irregular geometries, complex boundary conditions without requiring extensive empirical correlation databases. Furthermore, the particle-based nature of DEM allows for direct visualization and analysis of blast wave propagation mechanisms, offering insights into local loading patterns that are difficult to obtain through traditional empirical formulations.

2.3. Simulation design

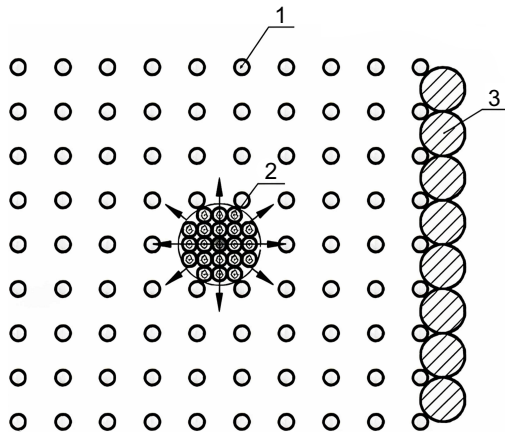


Fig. 1. Simulation model scheme: 1 – air particles system, 2 – explosive particles system, 3 – obstacles.

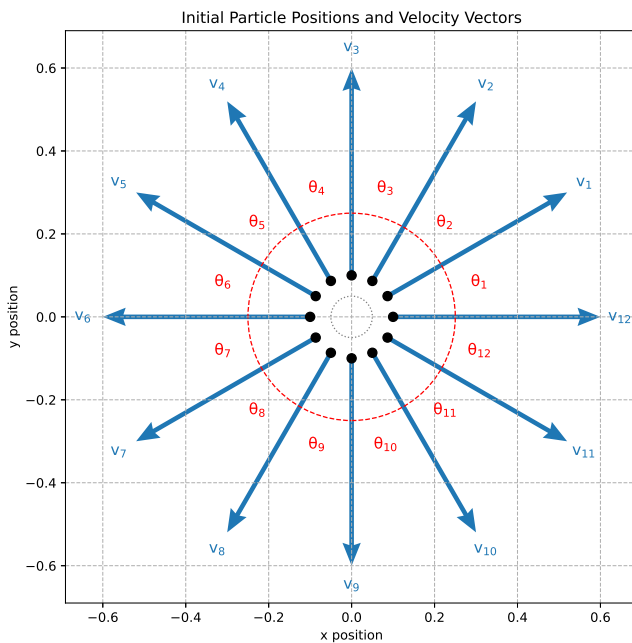


Fig. 2. Initial velocity vectors of 12 particles (as an example).

2.4. Collision modeling

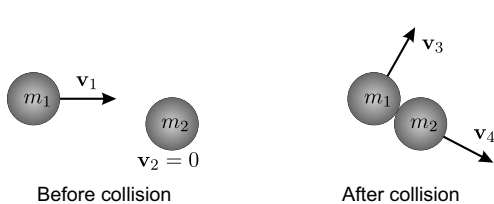


Fig. 3. Particle interaction.

before collision, particle m_1 moves with velocity $\mathbf{v}_1$, and particle m_2 may have $\mathbf{v}_2 = 0$ (or a small velocity); after collision, both particles acquire new velocities $\mathbf{v}_3$ and $\mathbf{v}_4$, determined by the conservation of momentum.

Such a model collision is usually considered as a conservative momentum transfer (within the chosen elasticity) with possible energy dispersion. In particular, in the case of a perfectly elastic collision, masses m_1 with velocity $\mathbf{v}_1$ and m_2 with velocity $\mathbf{v}_2$ satisfy the equality (3)

Particle model of explosion. In our simulation, explosions are modeled as a set of discrete particles (Figure 1) (presented in 2D space as circles) that fly radially from the center of the explosion. This approach simplifies the complex particle motion in explosions, preserving the main physics of impulse transfer.

The main idea of our simulation is to represent the explosive energy release as an instantaneous radial distribution of impulse. When an explosion occurs, a certain number of explosive particles are created at the detonation center, which receive initial velocities directed outward. The radial scattering of these particles mimics the pressure propagation in a real explosion process.

The algorithm for generating particles in an explosion for each of the n particles:

- Calculate the distribution angle:
 $\theta_i = \frac{2\pi \times i}{N}$, $i = \overline{1, n}$;
- Calculate the direction vector:
 $\mathbf{v}_i = (\cos(\theta_i), \sin(\theta_i))$, $i = \overline{1, n}$;
- Place the particle in the center of the explosion with a small offset in the direction of the vector.
- Give the particle an initial velocity (determined as a simulation parameter)
 $\mathbf{v}_i = (v_{xi}, v_{yi})$, $i = \overline{1, n}$
 along the vector direction (see Figure 2).

This approach ensures uniform distribution of explosive energy in all directions, corresponding to spherical (and in our 2D case – circular) propagation of the shock wave from a point source.

$$m_1 \mathbf{v}_1 + m_2 \mathbf{v}_2 = m_1 \mathbf{v}_3 + m_2 \mathbf{v}_4 \quad (3)$$

and kinetic energy is also conserved. If the collision is inelastic, some energy is lost in the form of heat or deformation. In both cases, the velocities after the collision directly affect the further distribution of impulses in the system and the dynamics of particle motion, simulating the transfer of explosive load in the environment.

Types of objects and collision processing mechanism. In our simulation, four main types of objects are used (see Figure 4), each with its own collision type identifier (*collision_type*):

ExplosionBall — explosive particles created at the epicenter and flying out with high initial velocity;

AirBall — air particles evenly distributed throughout the simulation space;

ModularPart — elements of the modular structure that receive a blow from the explosive wave;

Border — simulation space boundaries.

The Pymunk physics engine used in our simulation is based on the Chipmunk2D library [9] and uses the GJK/EPA (Gilbert–Johnson–Keerthi / Expanding Polytope Algorithm) [10] algorithm for determining collisions between complex shapes. The collision processing process consists of several stages: first, a fast initial check for possible collisions is performed using bounding rectangles (AABB) – so-called Broad Phase; then a detailed check for collisions is performed using the GJK/EPA algorithm (Narrow Phase); then forces and impulses are calculated that are transferred during the collision; finally, the velocities and positions of objects are corrected according to the calculated impulses.

For processing different types of collisions in the simulation, specialized handlers are implemented:

- **Collision of particles with the structure** — when an explosive particle or air particle collides with a structure element, the time of the first contact is determined, the collision angle is calculated, an event is registered in the corresponding module of the structure, and the size and mass of the explosive particle are scaled by 30%. In addition, a rebound impulse with the elasticity coefficient C_e is applied. If the particle becomes too small (radius $r < r_{min}$), it is removed from the simulation.
- **Collision between explosive particles and air particles** — in such contacts, the explosive particles are reduced in size and mass by 10%. This simulates the energy dispersion of the explosion through the environment.
- **General collision handler between particles** — applies a base impulse of magnitude I_b .
- **Exit beyond the simulation boundary** — particles that reach the boundaries of the simulation space are automatically removed from the model.

The system also implements collection and storage of data about each collision for further analysis. The information includes particle mass, velocity at the moment of collision, event time, collision angle (in radians and degrees), normal vector, structure module identifier, and the number of previous collisions with other objects. This data is exported in Excel table format, which allows for further analysis of impulse distribution along the structure, identification of maximum impact zones, and evaluation of the effectiveness of different protection configurations.

For correct interpretation of results in physical quantities, the following scaling ratio is applied:

$$1 \text{ meter} = 4 \text{ dimensionless units in simulation.} \quad (4)$$

This allows for relating simulation results to real physical parameters and comparing them with experimental data.

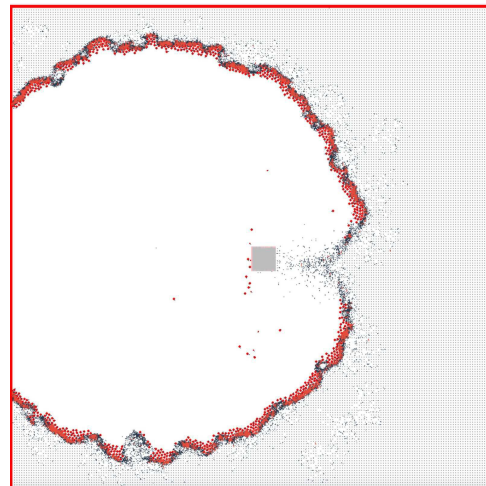


Fig. 4. Simulation interface screenshot: propagation of the explosive wave and interaction with the modular structure. Red points — explosive particles, dark blue — air, square — modular structure, red lines — simulation boundaries.

2.5. Particle parameters in the physical engine

In the Pymunk physics engine, particle parameters are characterized by a set of physical properties that form their behavior when interacting with the environment and other objects. Let us consider the theoretical aspects of the main parameters that determine the dynamics of particles in the simulation of an explosion.

Elasticity (Elasticity) determines the energy conservation characteristic during collisions. Theoretically, the elasticity value ranges from 0.0 to 1.0, where a zero elasticity corresponds to an absolutely inelastic collision, in which objects lose all kinetic energy, and maximum elasticity characterizes an absolutely elastic collision with full energy conservation. At intermediate values, some energy is preserved, and some is dispersed, imitating losses through deformation, heating, and sound effects. In the context of explosion modeling, elasticity reflects the ability of shock waves to preserve energy, gradually dispersing it through multiple interactions with obstacles and the environment.

Friction (Friction) is a parameter that characterizes the tangential resistance at the contact of objects. At the collision of two bodies, the friction force affects their ability to slide relative to each other at the contact point. Zero friction coefficient corresponds to ideal smooth surfaces, maximum creates significant sliding resistance. Friction significantly affects the energy distribution at oblique collisions, changing the shock wave propagation characteristics in the simulation space.

Mass (Mass) is a fundamental parameter that determines the object's inertia and its ability to transfer impulse during collisions. Particles with greater mass transfer a greater impulse at the same velocity, allowing for the simulation of different components of the explosion process. The ratio of mass between explosion particles and air is critical for the accurate reproduction of shock wave propagation.

Radius (Radius) determines the spatial characteristics of the particle and directly affects the collision mechanics. From the radius, the contact area at collisions, moment of inertia, and visual representation of the particle in the simulation depend. Reducing the radius at collisions allows for the simulation of energy dispersion with distance.

Moment of inertia is calculated based on mass and radius and determines the object's stability to rotation. For disk particles used in the simulation, the moment of inertia is calculated using the moment formula for a solid circle, ensuring physically correct behavior during rotational movements.

These parameters form a complex system of physical characteristics that allows for the simulation of a wide range of explosion scenarios. Their values can be adapted depending on specific research tasks or for calibrating the model to empirical data. The theoretical flexibility of these parameters makes the model a powerful tool for studying the physics of explosions in various conditions.

In the modeling process, it is important to consider the interdependence of these parameters. For example, changing elasticity affects the scattering of energy, which in turn may require other parameters to be adjusted to maintain physical consistency. Thus, theoretically justified parameter tuning allows for reproducing complex explosion physics with sufficient accuracy at relatively low computational costs.

2.6. Modeling structures

One of the main innovations of our approach is the modular representation of structures, allowing for detailed analysis of how the explosion impulse is distributed between different parts of the building. Such a granular structure gives the opportunity to obtain analytical data that is difficult to obtain in the case of monolithic models.

2.7. Modular approach

In our simulation, structures are not modeled as single rigid bodies, but as a collection of smaller modules. This is implemented through an hierarchical structure of classes. Each `ModularPart` is a basic building element that represents a separate discrete part of the structure. The module has a static body type (i.e., it is stationary but registers collisions), a square shape of 2 dimensionless units, a unique identifier that depends on its position, and a counter of collisions with color change upon damage (for visualizing load). The `ModularObstacle` class is responsible for creating and managing a

collection of ModularPart, organizing them in a grid structure. ConcreteSquare is a high-level class that forms concrete structures, abstracting low-level details of the model. Such an approach allows for detailed analysis of impulsive load, studying the difference in loads on different parts of the structure, as well as modeling gradual destruction by tracking damage accumulation. In addition, the modular nature provides flexibility in creating complex shapes and gives visual feedback due to color change of modules upon load increase.

2.8. Collection of data about impulse for each module

One of the key features of our model is the collection and analysis of data about impulse for each structural element separately, allowing for a deeper understanding of how the explosion wave affects different parts of the structure. Each ModularPart has its own impulse recording system; when a particle of the explosion collides with the structure, the system registers the fact of collision, identifies the corresponding structural element, registers the number of collisions, and calculates the impulse transferred to the object. This value is determined by the formula $I = m \times \sqrt{v_x^2 + v_y^2}$, where v_x and v_y are the components of the particle velocity. Using the Euclidean norm allows for considering the real particle velocity regardless of the angle of attack. Collected data are recorded in the Excel table for further analysis and contain the module identifier, collision time, particle type, mass, and collision angle. Such an approach allows for obtaining graphs of impulse-distance dependencies and evaluating structural vulnerability of structures.

The verification analysis and all subsequent calculations were performed using a custom Python script developed with the NumPy and SciPy libraries, ensuring reproducibility and accuracy.

3. Results

The final step of this study is model validation by verifying its results using comparison with data from the UFC 3-340-02 standard [3]. The purpose of verification is a quantitative assessment of the model's ability to reproduce the physical characteristics of the explosion impulse distribution at different distances from the source. Direct comparison of raw simulation data (i_{sim}), obtained as the sum of discrete elements' pulses, with the values of shock wave impulse from UFC (i_{ufc}) is not correct due to potential discrepancies in physical nature and measurement units of these quantities. Therefore, two alternative approaches for scaling simulation data were applied to enable comparison.

3.1. Linear scaling through proportional coefficient

The first approach involves using a simple linear coefficient k for scaling simulation data:

$$i_{scaled} = k \cdot i_{sim}. \quad (5)$$

The optimal value of this coefficient (k_{opt}), which ensures the best agreement between two data sets in the sense of minimizing the mean squared error (RMSE), was determined using the least squares method. This standard statistical method allows for finding the proportionality coefficient that minimizes the sum of squared differences between the dependent (UFC) and independent (scaled simulation) variables. The formula for calculating the optimal coefficient has the form:

$$k_{opt} = \frac{\sum(i_{sim} \cdot i_{ufc})}{\sum i_{sim}^2}, \quad (6)$$

where summation is performed over all distance points R , for which both values of the impulse are available. The calculated value was $k_{opt} \approx 0.00004343$. This coefficient was used as a tool for scaling simulation data to a common scale with UFC data for further analysis of verification error. On Figure 5 is presented a graphical comparison of the values of scaled impulses obtained using the described scaling method with data from the UFC document [3].

The results of comparison of the original data UFC with scaled simulation data using this method, as well as calculated absolute and relative errors that characterize the model's consistency with the standard, are given in Table 1.

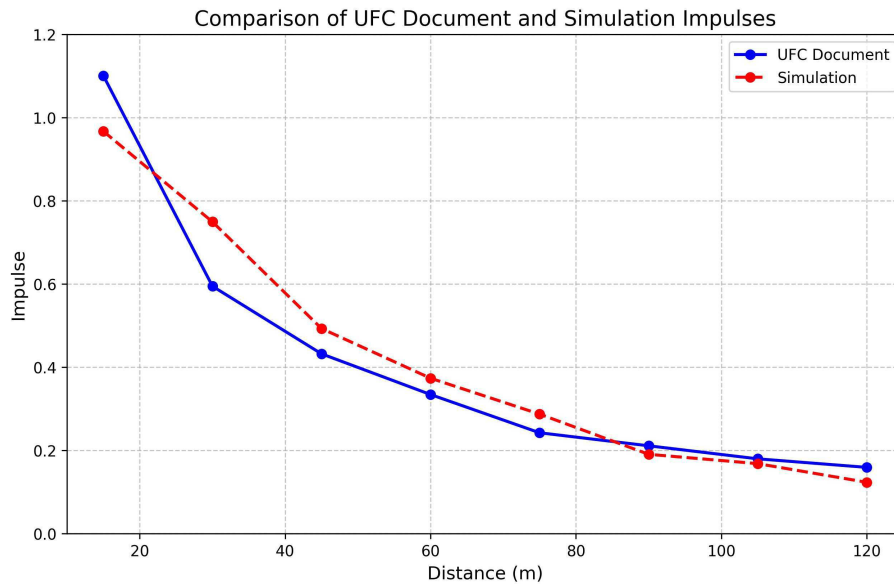


Fig. 5. Comparison graph of scaled impulse values from simulation and UFC data.

Table 1. Verification results using the proportional coefficient method:
Comparison of UFC 3–340–02 data and scaled simulation data for different distances.

R (m)	i_{ufc} (psiB·s)	i_{sim}	i_{scaled} (psiB·s)	E_{abs} (psiB·s)	$E\%$
15	1.10080	22267.44	0.95097	0.14983	13.61
30	0.59472	17258.76	0.73706	0.14234	23.93
45	0.43224	11353.49	0.48487	0.05263	12.18
60	0.33436	8602.54	0.36739	0.03303	9.88
75	0.24246	6616.28	0.28256	0.04010	16.54
90	0.21114	4390.75	0.18751	0.02363	11.19
105	0.17982	3873.75	0.16544	0.01438	8.00
120	0.15931	2842.02	0.12137	0.03794	23.81

3.2. Linear regression model with constant offset

An additional scaling method was considered, based on the full linear regression model. Unlike simple multiplication by a coefficient, this method involves using linear transformation of the form:

$$i_{scaled} = a + b \cdot i_{sim}, \quad (7)$$

where a and b are regression coefficients. Such an approach allows for not only proportionality between data but also possible systematic deviations. For determining optimal values of coefficients, the method of minimizing the sum of squared deviations between i_{scaled} and i_{ufc} was applied using formulas:

$$b = \frac{n \sum (i_{sim} \cdot i_{ufc}) - \sum i_{sim} \sum i_{ufc}}{n \sum i_{sim}^2 - (\sum i_{sim})^2}, \quad (8)$$

$$a = \frac{\sum i_{ufc} - b \sum i_{sim}}{n},$$

where n is the number of measurement points.

The parameters were optimized using the least squares method, which minimized the sum of squared deviations between predicted and actual values. Application of this classical statistical method allowed for finding optimal values of coefficients $a \approx -0.01696$ and $b \approx 0.00004392$.

The results of comparison of the original data UFC with scaled simulation data using this method, as well as calculated absolute and relative errors, are given in Table 2.

Table 2. Verification results using the linear regression method:
Comparison of UFC 3–340–02 data and scaled simulation data for different distances.

R (m)	i_{ufc} (psiB·s)	i_{sim}	i_{scaled} (psiB·s)	E_{abs} (psiB·s)	$E\%$
15	1.1008	22267.44	0.96094	0.13986	12.71
30	0.59472	17258.76	0.74098	0.14626	24.59
45	0.43224	11353.49	0.48164	0.04940	11.43
60	0.33436	8602.54	0.36083	0.02647	7.92
75	0.24246	6616.28	0.27360	0.03114	12.84
90	0.21114	4390.75	0.17586	0.03528	16.71
105	0.17982	3873.75	0.15316	0.02666	14.83
120	0.15931	2842.02	0.10785	0.05146	32.30

3.3. Power-law scaling model

Recognizing the inherently non-linear nature of blast wave energy dissipation with distance, a more physically sound power-law scaling model was investigated. This approach assumes a relationship of the form:

$$i_{scaled} = A \cdot (i_{sim})^B,$$

where A is a scaling coefficient and B is the power-law exponent. To determine the optimal parameters A and B , the equation is linearized by taking the natural logarithm of both sides:

$$\ln(i_{scaled}) = \ln(A) + B \cdot \ln(i_{sim}).$$

This transformed equation represents a linear relationship between $\ln(i_{sim})$ and $\ln(i_{scaled})$, where B is the slope and $\ln(A)$ is the y -intercept. A linear regression was performed on the log-transformed data to find these parameters.

The analysis yielded optimal parameters of $A \approx 0.000124$ and $B \approx 0.881$. Thus, the final conversion formula is:

$$i_{scaled} = 0.000124 \cdot (i_{sim})^{0.881}.$$

The results of applying this model are presented in Table 3.

Table 3. Verification results using the power-law scaling method.

R (m)	i_{ufc} (psiB·s)	i_{sim}	i_{scaled} (psiB·s)	E_{abs} (psiB·s)	$E\%$
15	1.1008	22267.44	0.8426	0.2582	23.5
30	0.5947	17258.76	0.6731	0.0784	13.2
45	0.4322	11353.49	0.4654	0.0332	7.7
60	0.3344	8602.54	0.3644	0.0300	9.0
75	0.2425	6616.28	0.2891	0.0466	19.3
90	0.2111	4390.75	0.2014	0.0097	4.6
105	0.1798	3873.75	0.1804	0.0006	0.3
120	0.1593	2842.02	0.1373	0.0220	13.8

3.4. Comparative analysis of scaling methods

To objectively determine the most effective scaling method, a comparative analysis was conducted using three key performance metrics: Mean Absolute Percentage Error (MAPE), Root Mean Squared Error (RMSE), and the Integral Area Error. MAPE provides a measure of the average error percentage, RMSE penalizes larger errors more heavily, and the Integral Area Error assesses the overall fit of the impulse–distance curve by comparing the area under the scaled simulation curve to the reference UFC curve, approximated using the trapezoidal rule.

The results for all three methods are summarized in Table 4.

Table 4. Comparative analysis of scaling methods.

Scaling Method	MAPE (%)	RMSE	Integral Error (%)
Proportional Coefficient	14.9	0.0794	5.2
Linear Regression	16.7	0.0788	3.6
Power–Law Model	11.4	0.0985	1.5

The analysis presented in Table 4 clearly demonstrates the superiority of the Power–Law Model. It is the most effective for three primary reasons. First, it yields the lowest Integral Area Error of only 1.5%, indicating the best overall agreement across the entire distance range. Second, it achieves the lowest Mean Absolute Percentage Error (MAPE) of 11.4%, signifying the highest average point-wise accuracy. Finally, from a theoretical standpoint, the power–law approach is the most physically justified, as it correctly models the inherent non-linearity of blast wave energy dissipation, unlike the linear assumptions of the other two methods. This combination of empirical accuracy and theoretical soundness confirms the power-law model as the optimal method for scaling the simulation results.

4. Discussion

4.1. Model limitations and simplifications

Although the results meet the established standards within the permissible errors, the model uses a number of simplifications. Two-dimensional nature of the simulation is one of the main limitations. Real explosions have a three-dimensional nature, and this model operates with circular explosion waves in two-dimensional space. This means that such effects as vertical pressure distribution, reflection of shock waves from complex three-dimensional geometries, shock wave diffraction around obstacles, formation of vortex structures in gases after explosion are not taken into account. Simplified representation of the explosion assumes instantaneous creation of particles with initial velocities, which does not reflect the complex chemical kinetics of detonation. Real explosions include the initial detonation phase, where a gas layer with high pressure is formed. In this model, this is omitted, as the main focus is on the impulse transfer, not on the internal explosion processes. Although the model takes into account the basic laws of impulse conservation, it does not reflect the nonlinear mechanical properties of materials under the action of explosion, structural deformation, and destruction, detailed explosion product distribution. Despite these simplifications, the results confirm the feasibility of the applied approach. The main goal of the model is to assess the impulse transfer from explosion to structures with adequate accuracy.

4.2. Future developments and enhanced physical realism

To address the inherent limitations of the 2D DEM approach and enhance model fidelity for complex blast scenarios, several critical developments are planned. While the current model demonstrates good agreement with UFC standards for impulse prediction, future enhancements will target specific physical phenomena essential for comprehensive blast analysis.

The extension to three-dimensional DEM formulations will capture spherical wave front geometry and out-of-plane pressure distributions currently absent in the 2D approach. This development will incorporate proper spherical wave propagation algorithms and enable modeling of complex 3D structural geometries, addressing the geometric limitations that restrict current applications to simplified planar configurations.

Implementation of wave reflection and diffraction algorithms represents a critical advancement for urban blast scenarios. The incorporation of Mach reflection processes, where incident and reflected shock waves interact to form triple points, will enable accurate prediction of enhanced pressure loading regions. These triple point configurations are particularly significant for structures with irregular geometries or in confined spaces where multiple reflections create complex pressure fields. The model will integrate algorithms for regular and irregular reflection patterns, including transition criteria between reflection modes based on incident angle and surface impedance properties.

Integration of explicit pressure–time history calculations will complement the current impulse-based approach. Future developments will incorporate pressure calculations based on accumulated particle interactions, enabling generation of Friedlander-type pressure profiles with both positive and negative phases. This enhancement will support direct comparison with experimental pressure measurements and provide peak overpressure values required for design calculations, addressing current limitations in pressure prediction capabilities.

Validation protocols will be expanded to include experimental blast data encompassing both pressure and impulse measurements at multiple scales, ensuring model reliability across the full range of engineering applications.

5. Conclusions

The study showed that the particle–based explosion wave simulation in Pymunk is an effective method for predicting blast loading on structures in a given distance range, however, it requires further development to improve the accuracy of the model. The obtained results confirm that the partially discrete approach correctly approximates the impulse transfer processes. The modular representation of structures allows for analyzing load distribution with high detail. The simulation is consistent with the UFC [3] standards, confirming its reliability. Future studies will focus on extending the model to calculations of positive phase explosion pressure and further model verification, including the implementation of three-dimensional DEM formulations, wave reflection and diffraction algorithms, and explicit pressure–time history calculations.

-
- [1] Chester A., Critchley R., Hazael R. A comparison of far-field explosive loads by a selection of current and emerging blast software. *International Journal of Protective Structures*. **16** (2), 387–418 (2024).
 - [2] Liu M. B., Liu G. R., Zong Z., Lam K. Y. Computer simulation of high explosive explosion using smoothed particle hydrodynamics methodology. *Computers & Fluids*. **32** (3), 305–322 (2003).
 - [3] UFC 3-340-02 Engineers. Structures to Resist the Effects of Accidental Explosions. Department of Defense, Washington, DC (2005).
 - [4] Friedlander F. G. The diffraction of sound pulses. I. Diffraction by a semi-infinite plane. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*. **186** (1006), 322–344 (1946).
 - [5] Shirbhate P. A., Goel M. D. A Critical Review of Blast Wave Parameters and Approaches for Blast Load Mitigation. *Archives of Computational Methods in Engineering*. **28**, 1713–1730 (2020).
 - [6] Lomazzi L., Giglio M., Manes A. Analysis of the blast wave – structure interface phenomenon in case of explosive events. *IOP Conference Series: Materials Science and Engineering*. **1038** (1), 012083 (2021).
 - [7] Ullah A., Ahmad F., Jang H.-W., Kim S.-W., Hong J.-W. Review of analytical and empirical estimations for incident blast pressure. *KSCE Journal of Civil Engineering*. **21** (6), 2211–2225 (2017).
 - [8] Baker W. E., Cox P. A., Westine P. S., Kulesz J. J., Strehlow R. A. *Explosion hazards and evaluation*. Elsevier (1983).
 - [9] Chipmunk2D. Chipmunk2D Physics: Official Documentation – Collision Detection (2023).
 - [10] Montaut L., Le Lidec Q., Petrik V., Sivic J., Carpentier J. Collision Detection Accelerated: An Optimization Perspective. Preprint arXiv:2205.09663 (2022).

Двовимірне моделювання імпульсів вибуху на споруди з використанням методу дискретних елементів

Мартинюк П. М., Кочкаров В. Д.

*Кафедра комп'ютерних наук та прикладної математики,
Національний університет водного господарства та природокористування,
вул. Соборна, 11, 79000, Рівне, Україна*

Вибухові явища становлять значну загрозу для структурної цілісності, що вимагає точних моделей прогнозування для проектування стійкої інфраструктури. Традиційні обчислювальні підходи, такі як CFD та FEM, хоча й детальні, вимагають значних обчислювальних ресурсів та спеціалізованої експертизи. Це дослідження представляє альтернативний підхід з використанням методу дискретних елементів (МДЕ), реалізованого через фізичний рушій Rumpk для 2D моделювання вибухів. Розроблений метод моделює вибухи як систему радіально розподілених частинок з початковими імпульсами, симулюючи поширення ударної хвилі через зіткнення частинок. Структури представлені з використанням модульного підходу, що дозволяє детальний аналіз розподілу імпульсу між різними елементами будівлі. Симуляція відстежує події зіткнень і обчислює передачу імпульсу, використовуючи принципи збереження моменту. Валідація моделі була виконана відповідно до стандартів UFC 3–340–02 шляхом дослідження трьох методів масштабування: пропорційного коефіцієнта, лінійної регресії та нелінійної степеневі моделі. Степенева модель продемонструвала найкращу узгодженість з еталонними даними, підтвердивши точність моделі із загальною інтегральною похибкою всього 1.5%. Цей обчислювально ефективний підхід забезпечує практичний інструмент для інженерів-конструкторів та міських планувальників для включення розгляду стійкості до вибуху без необхідності високопродуктивних обчислювальних ресурсів.

Ключові слова: *вибух; ударна хвиля; імпульс; фізичне моделювання; метод дискретних елементів.*