



HAL
open science

Windblown sand around obstacles -simulation and validation of deposition patterns

Nicolas Rosset, Régis Duvigneau, Adrien Bousseau, Guillaume Cordonnier

► **To cite this version:**

Nicolas Rosset, Régis Duvigneau, Adrien Bousseau, Guillaume Cordonnier. Windblown sand around obstacles -simulation and validation of deposition patterns. Proceedings of the ACM on Computer Graphics and Interactive Techniques, In press, 7 (1), pp.13. 10.1145/3651284 . hal-04566241

HAL Id: hal-04566241

<https://inria.hal.science/hal-04566241>

Submitted on 2 May 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

Windblown sand around obstacles – simulation and validation of deposition patterns

NICOLAS ROSSET, Inria Université Côte d’Azur, France
REGIS DUVIGNEAU, Inria Université Côte d’Azur, France
ADRIEN BOUSSEAU, Inria Université Côte d’Azur, France
GUILLAUME CORDONNIER, Inria Université Côte d’Azur, France

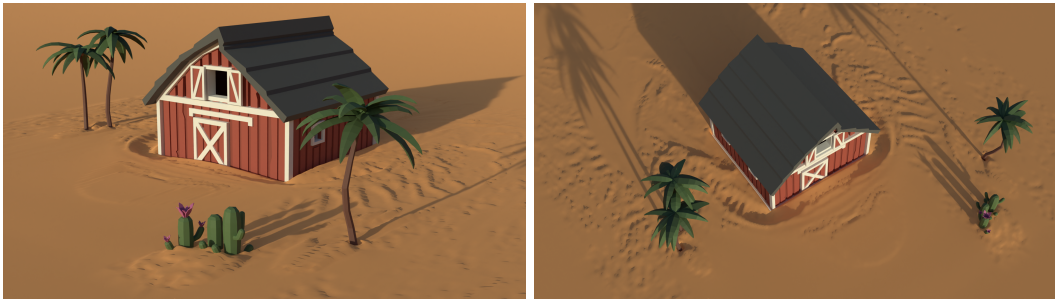


Fig. 1. Desertscape simulated with our method. Note the erosion and deposition patterns around the barn, trees, and cactus. 3D models of farm buildings and trees from quaternius.com under a CC0 license.

Sand dunes are iconic landmarks of deserts, but can also put human infrastructures at risk, for instance by forming near buildings or roads. We present a simulator of sand erosion and deposition to predict how dunes form around and behind obstacles under wind. Inspired by both computer graphics and geo-sciences, our algorithm couples a fast wind flow simulation with physical laws of sand saltation and avalanching, which suffices to reproduce characteristic patterns of sand deposition. In particular, we validate our approach via a qualitative comparison of the erosion and deposition patterns produced by our simulator against real-world patterns measured by prior work under controlled conditions.

CCS Concepts: • **Computing methodologies** → **Physical simulation**.

Additional Key Words and Phrases: Fluid simulation, Sand dunes, Sand deposition, Landscape

ACM Reference Format:

Nicolas Rosset, Regis Duvigneau, Adrien Bousseau, and Guillaume Cordonnier. 2024. Windblown sand around obstacles – simulation and validation of deposition patterns. *Proc. ACM Comput. Graph. Interact. Tech.* 7, 1 (May 2024), 13 pages. <https://doi.org/10.1145/3651284>

Authors’ addresses: Nicolas Rosset, Inria Université Côte d’Azur, France, nicolas.rosset@inria.fr; Regis Duvigneau, Inria Université Côte d’Azur, France, regis.duvigneau@inria.fr; Adrien Bousseau, Inria Université Côte d’Azur, France, adrien.bousseau@inria.fr; Guillaume Cordonnier, Inria Université Côte d’Azur, France, guillaume.cordonnier@inria.fr.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.

2577-6193/2024/5-ART \$15.00

<https://doi.org/10.1145/3651284>

1 INTRODUCTION

Around $\frac{1}{15}^{th}$ of the Earth's lands is covered by sand [L. Raffaele 2019], giving rise to fascinating dune landscapes, but also to unintended deposition of sand on infrastructures (buildings, roads, railways). Simulating the formation of dunes has attracted attention in computer graphics to create plausible virtual worlds [Paris et al. 2019], as well as in geo-sciences to anticipate and mitigate real-world sand deposition [Raffaele et al. 2022]. On the one hand, methods from computer graphics offer high speed and user control, but make simplifying assumptions that are seldom validated against real-world experiments [Taylor and Keyser 2023]. On the other hand, methods from geo-sciences achieve high accuracy by accounting for multiple physical effects, but these effects are costly to simulate [Andrea Lo Giudice and Preziosi 2020].

In this work, we draw inspiration from both communities to propose a simulator of sand erosion and deposition that focuses on *saltation* – the phenomenon of sand particles hopping over the terrain under the action of the wind, which dominates sand deposition patterns [Kok et al. 2012]. Our method relies on an efficient fluid simulator to model the 3D+t wind field over the terrain, coupled with a sand transport model derived from geo-science laws that only acts at the surface of the terrain. While simpler than advanced simulators from the engineering literature, our approach better accounts for wind flow around obstacles than prior work in computer graphics (Figure 1).

We validate our approach by simulating windblown sand around controlled configurations of obstacles, demonstrating a qualitative agreement with reference deposition patterns that have been recently measured in the real world [Poppema, Wijnberg, et al. 2022].

2 RELATED WORK

Sand erosion and deposition. Sand dunes shape through complex interactions between the sand bed and the air flowing above it, referred to as *aeolian processes* [Nichols 2009]. Intuitively, wind erodes the sand bed where its intensity is high, and it deposits the eroded particles where its intensity is low. Several modes of transportation of the eroded particles can be considered [Bagnold 1941], including *reptation* and *creep erosion* where heavy particles hop or roll on the ground over a few centimeters, *suspension* where light particles are carried by the wind over kilometers, and *saltation* where medium particles hop on the ground over distances of a few meters. Note however that there is no strict separation between these modes, and that some studies adopt a definition of saltation that includes reptation and creeping [Kok et al. 2012]. Once deposited, sand particles are also subject to *avalanching*, a process that typically triggers when the sand bed surface reaches a critical angle. Our method focuses on saltation since it is responsible for the majority of the sand mass transported by wind over short distances. Similar phenomena of erosion and deposition also happen for snow transport [G. E. Liston 1998; Schneiderbauer and Prokop 2011; Schneiderbauer, Tschachler, et al. 2008; Vionnet et al. 2014].

Aeolian sand transport simulation. The phenomena listed above can be simulated by coupling a fluid simulator (for the wind) and a transport simulator (for the sand). Andrea Lo Giudice, Nuca, et al. [2019], Tominaga [2018] and Zhou and T. Zhang [2023] survey methods to simulate sand and snow transport based on computational fluid dynamics (CFD). Closest to our problem setting is the work by Andrea Lo Giudice and Preziosi [2020], who propose a multiphase model to accurately simulate multiple phenomena induced by the mutual interactions of sand, air, and obstacles. However, accounting for all these phenomena within a volumetric simulation results in prohibitive computation time, up to four days on a 10-processor cluster for a single simulation. In contrast, we adopt here an efficient fluid simulator and a simplified sand transport model to achieve qualitatively similar deposition patterns in a few minutes (Sec. 4).

Instead of simulating macroscopic physical laws, methods based on cellular automata reproduce physical phenomena by implementing a small set of stochastic rules that dictate how sand is transported across neighboring cells of a grid discretization of the sand bed [O. Rozier 2013]. While early methods only consider sand dunes [Narteau et al. 2009; Werner 1995], recent works propose additional rules to model the effect of simple obstacles [Poppema, Baas, et al. 2022], or adapt the rules to model snow transport [Kochanski et al. 2019]. Since these methods model the impact of obstacles on wind and sand transport using a phenomenological point of view, each new effect requires developing dedicated rules. In contrast, we developed our method around physical laws, such that sand patterns emerge from a simple coupling between sand transport and wind dynamics.

The computer graphics community has proposed several approaches to simulate sand or snow transport with fast, albeit approximate algorithms. Early methods focus on the formation of ripples, as well as simple deposition and shadowing around obstacles [Benes and Roa 2004; Onoue and Nishita 2000; Wang and Hu 2012]. More recently, Paris et al. [2019] propose a fast aeolian simulation tailored to the interactive modeling of desert landscape. Key to their approach is a simplified model of fluid flow, which consists in warping a uniform wind field to align with the local slope of the terrain, along with erosion and deposition rules akin to cellular automata. This model targets the fast generation of large deserts (several hundreds of meters wide), but it does not reproduce the smaller-scale wind patterns we target, such as vortices that appear behind obstacles and that are responsible for characteristic sand erosion and deposition patterns. Taylor and Keyser [2023] extended this method to model interactions with obstacles and propose a GPU implementation for real-time simulation, to which we compare in Sec. 4. We differ by adopting a generalist and more realistic, albeit more expensive, fluid simulator. Our approach also relates to the work of Feldman and O'Brien [2002], who simulate wind flow and snow transport in a 3D grid surrounding obstacles. Similarly, we simulate a volumetric wind field to capture intricate flow patterns around obstacles, but we rely on a 2D grid to model sand saltation at the surface of the terrain.

Importantly, while most prior works in computer graphics were validated by comparing the overall shape of simulated dunes with photographs of real dunes [Paris et al. 2019], we conduct a more systematic experiment where we qualitatively compare our simulations on specific obstacle configurations for which ground truth deposition patterns have been measured [Poppema, Wijnberg, et al. 2022].

3 SIMULATING WINDBLOWN SAND

We now describe our model to simulate sand transport and deposition under the effect of wind. While we seek for a simple model, we build it on laws derived from geo-science experiments.

3.1 Sand transport

As mentioned in Sec. 2, sand transport under wind is dominated by saltation and avalanches [Kok et al. 2012]. Most previous works in geo-sciences [O. Rozier 2013] and computer graphics [Paris et al. 2019; Taylor and Keyser 2023] model sand as a height field and rely on proxies to simplify the wind dynamics at the sand surface. As an exception, Andrea Lo Giudice and Preziosi [2020] considers a fully coupled 3D transport of both the wind and the sand. While we take inspiration from this coupled model by simulating 3D fluid dynamics, we note that saltation and avalanches occur at the surface of the terrain, making these phenomena amenable to a 2.5D representation that tracks the evolution of the sand thickness h over the terrain. In what follows, we allow for a non-flat ground of altitude g , and define the surface altitude as $f = h + g$.

In computer graphics, transport of sand [Paris et al. 2019] or snow [Cordonnier et al. 2018] is simplified as directly linked to the wind velocity: sand is eroded by strong wind, and is deposited when the wind weakens or is shadowed by obstacles. While this approach is easily controllable

and fits well with procedural wind fields, it fails to capture the complex erosion and deposition patterns induced by physically-based 3D wind flows. Instead, we propose a formulation based on a sand transport rate \mathbf{Q} (unit: kg/m/s), which readily expresses the evolution of the sand thickness through volume conservation:

$$\frac{\partial h}{\partial t} + \frac{1}{\rho} \nabla \cdot \mathbf{Q} = 0, \quad (1)$$

where ρ is the sand density. Intuitively, if we consider a small volume of sand, the sand transport rate \mathbf{Q} expresses the mass of sand that flows through the bound of the volume, and its divergence gives the net balance in sand mass per time unit.

We split the sand transport rate into two terms: $\mathbf{Q} = \mathbf{Q}_s + \mathbf{Q}_a$, where \mathbf{Q}_s accounts for the sand moved by saltation, while \mathbf{Q}_a models the sand moved by avalanching. G. Strypsteen [2021] compares several formulations of saltation rate \mathbf{Q}_s and suggests that the following law by Zingg [1952] is a good fit to measured data:

$$\mathbf{Q}_s = \begin{cases} k_s \|\mathbf{u}_*\|^2 \mathbf{u}_*, & \text{if } h > 0 \\ 0, & \text{otherwise,} \end{cases} \quad (2)$$

where k_s depends on the grain size; we use $k_s = 0.13 \text{ kg.s}^2 \text{ m}^{-4}$ which corresponds to a median grain size of 1.3 mm , and \mathbf{u}_* is the shear velocity of the wind at the sand surface. The shear velocity is used in fluid mechanics to characterize the flow velocity profile at the wall boundary, and therefore represents here the capacity of the flow to entrain sediments.

We note that the wind velocity is null at the ground altitude, and therefore, linearizing the velocity profile at the vicinity of the ground gives the following relationship:

$$\mathbf{u}_* = \frac{1}{50\epsilon} \mathbf{u}_t (f + \epsilon), \quad (3)$$

where $\epsilon = 10^{-1} \text{ m}$ is the distance to the flow surface, and we obtained the coefficient $1/50$ by comparing our velocities with the velocities and shear velocities reported by G. Strypsteen [2021] from real measurements.

Sand is a granular material and as such flows down when the surface slope exceeds a critical *Coulomb angle* θ_{cr} . We express this *avalanching* process in the avalanche rate \mathbf{Q}_a . We set $\mathbf{Q}_a = 0$ if $h = 0$, otherwise we follow a rotation-invariant expression from geosciences [Lo Giudice et al. 2019]:

$$\mathbf{Q}_a = k_a \nabla f \frac{\max(\|\nabla f\| - \tan \theta_{cr}, 0)}{\|\nabla f\| \sqrt{1 + \|\nabla f\|^2}}. \quad (4)$$

The constant k_a depends on the adhesive properties of the sand but is poorly constrained [Lo Giudice et al. 2019; Andrea Lo Giudice and Preziosi 2020]. Compared to the time scale of saltation, avalanches are much faster and can be considered at equilibrium. Therefore, we assign a specific timestep to this phenomenon and set $k_a dt_a = 0.24 \rho dx^2$ – a compromise between numerical stability and fast convergence to equilibrium.

Discretization. We use a 2D staggered grid embedded in the heightmap (Fig. 2). We store the elevations at the centers of the cells (e.g., $h(i, j)$) for a cell centered at position (i, j) , while we store the transport rates and shear velocity at the interfaces between cells (e.g., $Q^x(i + 1/2, j)$ or $Q^y(i, j + 1/2)$). The surface gradient ∇f arising in the avalanche term naturally complies with this discretization, but the norms $\|\mathbf{u}_*\|$ and $\|\nabla f\|$ require interpolating the vector components to obtain their values at the flux locations. Note that when the surface is not flat, the tangential velocity (and hence the shear velocity) has a vertical component. This

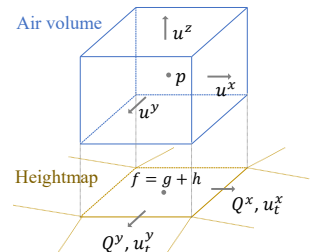


Fig. 2. Notations.

component is discarded in the definition of the transport rate but is still required for the computation of the norm of the shear velocity and therefore stored in the center of the cells.

The cases $Q_s = 0$ and $Q_a = 0$ when $h = 0$ require checking the presence of sand at the cells interface, while h is stored at the center. Furthermore, a largely divergent transport rate where h is small can lead to a negative sand thickness. To solve both issues, we consider independently each component of the transport rate, stored at a cell interface, e.g., $Q^x(i + 1/2, j)$. We want to limit the amount of material transported by the amount of material available in the incoming cell, called the *upwind* cell. We determine the cell *upwind* to an interface from the sign of the transport rate component stored on that interface (e.g., (i, j) if $Q^x(i + 1/2, j) > 0$), and denote by the subscript ${}^*_{up}$ a quantity stored in the upwind cell. Finally, we adjust the transport rate to the maximum amount that can be transported from the cell *upwind* during a time dt : $Q_{max} = h_{up} \frac{\rho dx}{dt N_{up}}$ where $N \in \{1, \dots, 4\}$ is, for each cell, the number of interfaces where the sign of the transport rates indicates that the sand leaves the cell. The final transport rate is then given component-wise ($i \in \{x, y\}$) by:

$$Q^i = \text{clamp}(Q_s^i + Q_a^i - Q_{max}^i, Q_{max}^i) \quad (5)$$

3.2 Wind flow

We assume that the wind is a Newtonian fluid and follows the incompressible Navier-Stokes Equation:

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = -\frac{1}{\rho} \nabla p + \nu \Delta \mathbf{u}, \quad (6)$$

where $\rho = 1.293 \text{ kg m}^{-3}$ is the air density and $\nu = 1.5 \cdot 10^{-5} \text{ m}^2 \text{ s}^{-1}$ is the air viscosity. We add the incompressibility condition

$$\nabla \cdot \mathbf{u} = 0. \quad (7)$$

We do not account for turbulence phenomena, which could be present in real life experiments, due to their complexity of implementation. Nevertheless, turbulence mainly acts as additional diffusion effects, which are inherently included in numerical diffusion.

We set the boundary conditions $\mathbf{u} = 0$ at the sand-air interface and we force a constant air inflow $\mathbf{u} = u_{inflow}$ on the front, left, right and top sides of the domain. We leave an open boundary condition at the back side, with Neumann boundary conditions for the velocity and Dirichlet conditions ($p = 0$) for the pressure.

We use Stable Fluid's [Stam 1999] Helmholtz decomposition: we first advect the velocity with an RK3 semi-lagrangian scheme, then we compute a pressure field that projects back the residual velocities into the divergence-free space. The Poisson problem arising from the pressure projections requires additional Neumann boundary conditions at the interface with the sand and the inflow domain bounds: $\mathbf{n} \cdot \nabla p = 0$, where \mathbf{n} is the normal to the obstacle or to the fluid domain. We use a sparse preconditioned conjugate gradient method for the pressure projection. Specifically, we run it with a threshold of $t = 10^{-6}$ and a maximum number of 6000 iterations as our convergence parameters.

Sand-wind interactions. We store the wind velocity and pressure in a 3D staggered grid (Fig. 2), with each three components of the velocity stored on the faces of the cell and pressure at the center of the cells. We align the 2D staggered grid of transport rates with the 3D staggered grid of wind velocities. Indeed, the transport rate for saltation depends on the shear velocity, itself depending on the wind velocity tangent to the sand surface. To compute the tangent velocity, we use a vertical linear interpolation of the wind velocity to fetch it close to the sand surface ($\mathbf{u}(f + \epsilon)$), and we deduce the tangential velocity with $\mathbf{u}_t = \mathbf{u} - \mathbf{n}(\mathbf{u} \cdot \mathbf{n})$. Note that non-flat sand surfaces lead to a

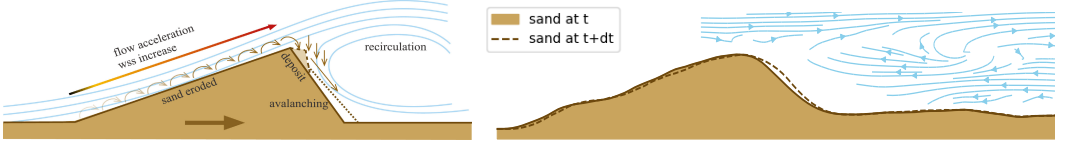


Fig. 3. Evolution of a transverse dune, schematically (a, from [Andrea Lo Giudice and Preziosi \[2020\]](#)) and simulated by our method (b). Note that we produced this figure by running a 3D simulation and extracting a 2D slice of the sand bed and wind field, yielding more complex vortex patterns than the 2D idealized case. Our method (b) aligns well with the idealized scenario (a) by eroding sand on the left side of the dune, which faces the wind, and by depositing sand on the right side of the dune, which faces the recirculation area where the wind forms a vortex.

vertical tangential velocity component, which is stored at the center of the 2D staggered grid. It might be confusing to handle this vertical component because the saltation transport rate Q_s is inherently 2D and only embeds the two horizontal components, but the computation of Q_s in Eq. 2 includes the norm of the 3D shear velocity which does include a vertical component.

The sand surface itself acts as a boundary condition for the wind. Encoded in a heightmap, the altitude values freely vary vertically and do not conform to a specific cell or intercell altitude. More generally, handling non-regular domains is a common problem in computer graphics, and we follow [\[Ng et al. 2009\]](#) who estimate the proportion α of the cell's face that intersects with the free air volume. Following a finite-volume approach, they scale the pressure gradients and the velocity by these proportions, effectively changing the equation for pressure projection to:

$$\nabla \cdot \alpha \nabla p = -\frac{\rho}{dt} \nabla \cdot \alpha \bar{\mathbf{u}}, \quad (8)$$

where α is the free fluid proportion and $\bar{\mathbf{u}}$ is the intermediate velocity after advection. We also set the velocity to $\mathbf{u} = 0$ when the associated face is entirely below the surface ($\alpha = 0$).

3.3 Summary of our algorithm

In summary, our algorithm starts with an initial heightmap with ground altitude g and sand thickness h , and velocities initialized at $\mathbf{u} = 0$ except for the prescribed inflow values. Then, we iterate over the following steps:

- Compute the proportions of the cell interface α intersecting with the fluid domain above the sand surface $f = h + g$.
- Project $\bar{\mathbf{u}}$ in the divergence-free space by solving Eq. 8. This results in the wind velocity \mathbf{u} .
- Fetch \mathbf{u} close to the sand surface f (altitude $f + \epsilon$), deduce the tangential velocity \mathbf{u}_t and then the shear velocity \mathbf{u}_* from Eq. 3.
- Compute the saltation transport rate (Q_s , Eq. 2) and the avalanche transport rate (Q_a , Eq. 4), compute the component-wise maximum flux Q_{max} and deduce the final transport rate Q from Eq. 5.
- Evaluate erosion/deposition from mass conservation (Eq. 1) and update the sand thickness h .

In practice, we let the fluid flow simulation run for 100 iterations before starting the erosion and deposition processes.

4 RESULTS

We validate our algorithm by reproducing sand deposition patterns documented in prior work. We start with idealized patterns for which we have reference diagrams or simulations, followed by real-world patterns for which we have reference measurements.

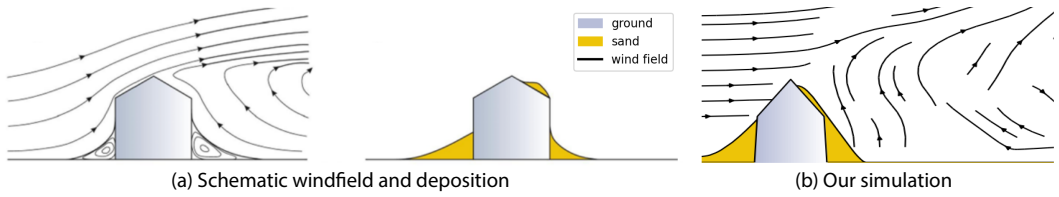


Fig. 4. Windfield and sand deposition over a gable roof, schematically (a, from [L. Raffaele \[2019\]](#)) and simulated by our method (b). Note that we produced this figure by running a 3D simulation and extracting a 2D slice of the sand bed and wind field.

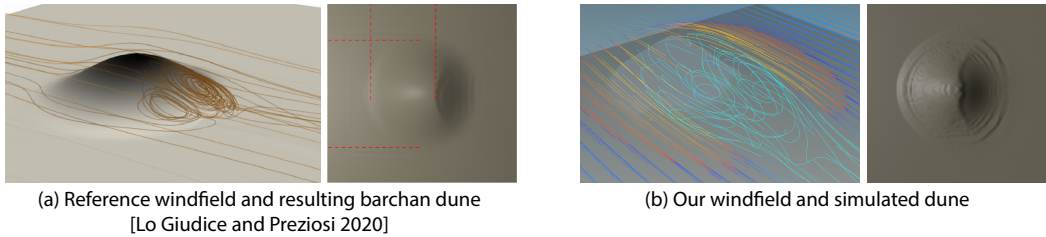


Fig. 5. Formation of a barchan dune by blowing wind on a Gaussian-shaped pile of sand. Reference simulation by [Andrea Lo Giudice and Preziosi \[2020\]](#)(a) and our simulation (b). Our method produces similar results yet employs a faster fluid solver and only considers saltation and avalanching.

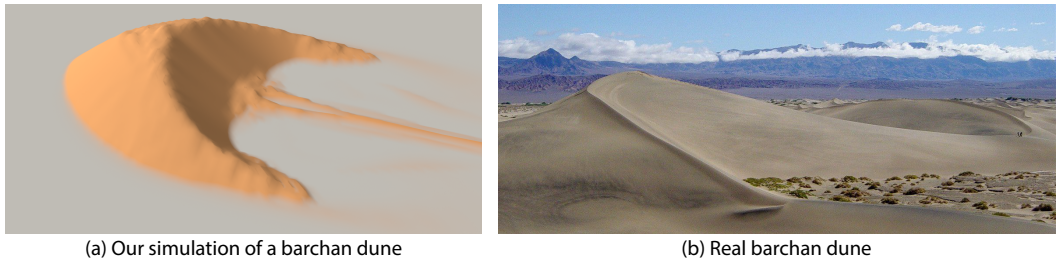


Fig. 6. Visual comparison between our simulation (a) and the picture of a real barchan dune (b, Mesquite Sand Dunes in the Death Valley by Daniel Mayer, Wikimedia Commons).

4.1 Idealized deposition patterns

Transverse dune. Prior work describes how transverse dunes exhibit a characteristic triangle profile that moves in the direction of the wind as sand gets eroded on the side facing the wind, and gets deposited on the side facing away from the wind, where wind recirculation occurs [[Bruno and Fransos 2015](#); [Andrea Lo Giudice and Preziosi 2020](#)]. Figure 3a illustrates this configuration schematically. Figure 3b shows that running our simulator on a triangle-shaped dune for a single time step produces a similar behavior.

Gable roof. Predicting sand deposition over human-made structures is of critical importance to preserve such structures. [[L. Raffaele 2019](#)] report typical sand deposition patterns around common structures, including gable roofs (Figure 4(a)). Our method produces similar patterns on a simple house, with sand deposition against the front and back wall, as well as over the back side of the roof (Figure 4(b)).

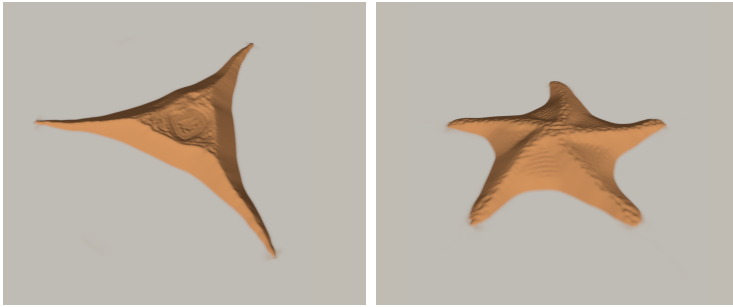


Fig. 7. Formation of star dunes by blowing wind on a truncated cone of sand, alternating over three (left) and five (right) wind directions. We refer to Fig2a) and Fig2c) from [D. Zhang et al. \[2012\]](#) for a comparison.

Barchan dune. Barchans are crescent-shaped dunes that commonly appear in sand deserts all over the world [[Courrech du Pont 2015](#)]. [Andrea Lo Giudice and Preziosi \[2020\]](#) reproduced such dunes by initializing their high-end simulator with a Gaussian-shaped pile of sand under a strong uni-directional wind. After a few time steps, the crescent shape emerges. Figure 5 reports their visualizations, along with the results of our simulator. While approximate, our fast method reproduces well the characteristic shape of the barchan. Figure 6 illustrates the result of our simulation when we let it run longer, along with the picture of a real barchan dune for visual comparison.

Star dune. [[D. Zhang et al. 2012](#)] used cellular automata to simulate the formation of star dunes. In their experiments, star dunes emerge from a truncated conical sand pile when wind alternates over several directions. They observed that star arms form in the presence of an odd number of wind directions, the number of arms being equal to the number of directions. We reproduced their experiment for wind alternating over three and five directions, and obtained similar star-shaped patterns, as shown in Figure 7. However, because this experiment amounts to simulate wind flow over multiple, alternating expositions, it requires more iterations and computation time than our other experiments (Table 1).

4.2 Real-world patterns

We now compare our results to sand deposition patterns measured in real-world experiments. In particular, we build on the extensive series of experiments conducted by [Poppema, Wijnberg, et al. \[2022\]](#), who measured sand deposition patterns around cuboid shapes placed under diverse configurations (spacing, orientation with respect to the wind) on a windy sand beach in the Netherlands.

We reproduced these configurations with our simulator and provide a visual comparison to their measurements in Figure 8. We adjusted the wind speed and viscosity of our simulator to best match their measures. While qualitative, this evaluation shows that our simulator reproduces well the erosion and deposition pattern around obstacles, as a function of the spacing between the obstacles and the inclination of the wind.

We also use measurements from [Poppema, Wijnberg, et al. \[2022\]](#) as a reference to compare our method to the real-time simulator of [Taylor and Keyser \[2023\]](#), which extends the work of [Paris et al. \[2019\]](#) to handle obstacles (see Figure 9). This simulator is based on stochastic rules that erode sand in front of the obstacle and deposit sand behind the obstacle. As such, this simulator cannot reproduce the complex erosion and deposition patterns observed in the real-world experiment,

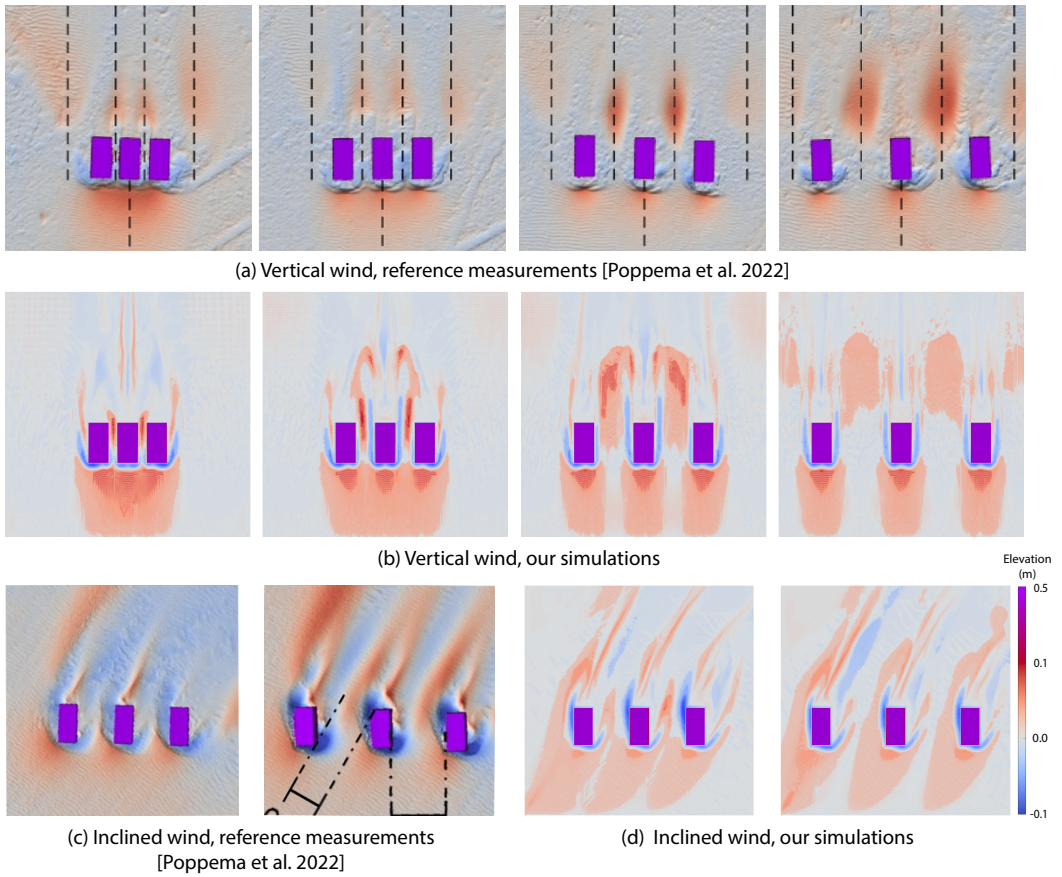


Fig. 8. Comparison to real-world measurements by Poppema, Wijnberg, et al. [2022] for different configurations of cuboids. Blue denotes erosion and red denotes deposition. Our method faithfully reproduces sand erosion and deposition patterns for cuboids with varying spacing under a vertical wind (a,b) and inclined wind (c,d). Note in particular how sand is eroded in front and around the cuboids, and deposited behind and between the cuboids as they are more spaced.

where the wind flow wraps around the obstacle, creating arc-shaped sand ripples in front and on the side of the cuboid. On the contrary, our method is able to reproduce such patterns.

4.3 Impact of resolution

The spatial accuracy of our simulator depends on the resolution of the underlying heightmap and volumetric grid. Fig 10 (top) visualizes barchan dunes obtained by simulating unidirectional wind over a Gaussian pile of sand, on spatial domains of increasing resolution. The characteristic crescent shape of the dune emerges in all cases, but the ridge of the dune becomes sharper at higher resolution. Fig 10 (bottom) shows the result of simulating a thin bed of sand under unidirectional wind around a pyramid, at different resolutions. At higher resolution, the sand deposition in front of the obstacle and in the wake of the flow is more detailed. But high resolution comes at an increased computational cost, as reported in Table 1 and discussed next.

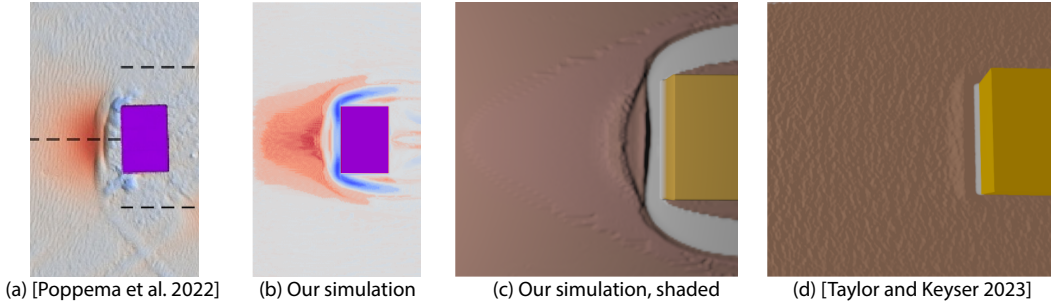


Fig. 9. Comparison to the real-time method by Taylor and Keyser [2023]. The measurements by Poppema, Wijnberg, et al. [2022] provide a reference of how sand erodes and deposits around a cuboid (a). Our simulator reproduces well this pattern (b,c), where sand level increases progressively in front of the object, then drops quickly and rises again to form a thin secondary dune, before dropping again to form a crescent-shaped cavity that wraps around the cube. In contrast, the procedural rules of Taylor and Keyser [2023] only produce a straight dune and cavity parallel to the front face of the cube (d, reproduced from Taylor and Keyser [2023]). Note that we used a lower initial sand thickness in (c) to better match the conditions in (d).

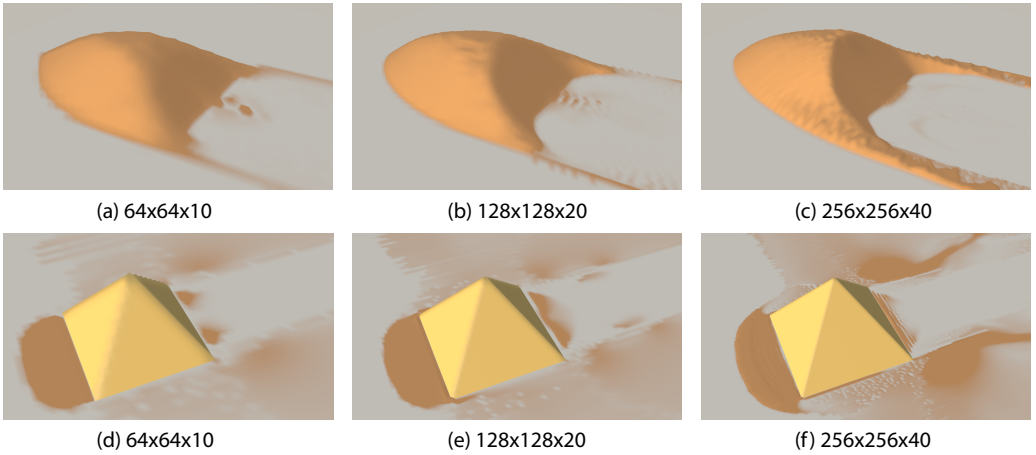


Fig. 10. Rendering of the simulations at different resolutions of a barchan dune emerging from a gaussian stack of sand (a, b, c), and of a thin layer of sand blown of a pyramid (d, e, f). While the different resolutions yield a similar pattern of depositions overall, higher resolution allows to capture sharper details.

4.4 Time performance

We implemented our algorithm in Python, using PyTorch for easy GPU integration. We provide our source code at https://gitlab.inria.fr/nrosset/windblown_sand. All timings are reported from a desktop computer equipped with a single NVIDIA RTX 3070 GPU with 8 GB dedicated memory. We ran experiments on domains with varying discretizations – more complex scenes requiring higher resolutions (Table 1). All simulations shown in this paper took between a few seconds to a few minutes, depending on the size of the grid and the number of iterations, which is orders of magnitude faster than the complex simulators used in engineering [Andrea Lo Giudice and Preziosi 2020]. Since all the erosion/deposition updates are made on a 2D grid, the bottleneck for our time performances is the 3D fluid simulation, and in particular, the pressure projection that requires the

experiment	resolution	# of iterations	time
Fig 3	128*128*30	150	0min14s
Fig 4	128*128*30	3500	5min32s
Fig 5	128*128*30	150	0min14s
Fig 6	128*128*30	600	1min14s
Fig 7	256*256*30	72000	55min30s
Fig 8	256*256*30	1300	9min10s
Fig 10 (a)	64*64*10	200	0min18s
Fig 10 (b)	128*128*20	400	0min50s
Fig 10 (c)	256*256*40	800	5min58s
Fig 10 (d)	64*64*10	120	0min11s
Fig 10 (e)	128*128*20	240	0min28s
Fig 10 (f)	256*256*40	440	3min02s

Table 1. Timings for the main results in this paper.

inversion of a matrix. The number of iterations here specified is the number of times we performed this projection during a given simulation.

5 LIMITATIONS

In addition to missing long-range sand transport modes (such as suspension in higher altitudes), the main limitations of our simulator stem from our representation of the ground as a heightfield defined over a grid. This representation is better suited to represent axis-aligned shapes, such as the cuboid in Figure 9, than curved shapes that suffer from aliasing artifacts. Furthermore, heightfields cannot represent obstacles with overhangs, like arches and fences, while such obstacles can form effective shielding structures [Bruno, Coste, et al. 2018].

6 CONCLUSION AND FUTURE WORK

We introduce a simulator that combines the physical phenomena responsible for windblown sand deposition patterns. Specifically, a fluid flow simulator captures the wind dynamics over a heightfield, and drives the transportation of sand through a process called saltation. We derive the erosion and deposition of sand from the balance of saltation and of another transport mechanism, avalanching, that dominates when the slope of the sand surface exceeds a critical angle. Our experiments demonstrate that our method can reproduce reference patterns from alternative methods and field studies, and capture both the transient emergence of dunes and the static buildup of typical sand patterns around obstacles. While the simplicity of our simulator makes it faster than the ones used in engineering, it does not yet reach the speed necessary for the interactive design of obstacles for sand mitigation measures [Raffaele et al. 2022]. Future work could rely on our simulator to train fast data-driven surrogate models suitable for inverse design applications [Rosset et al. 2023].

ACKNOWLEDGEMENTS

We thank Émilie Yu for helping with the figures. We also thank Jingwei Tang for implementing the fluid solver. This work was supported by the Agence Nationale de la Recherche project Invterra ANR-22-CE33-0012-01 and research and software donations from Adobe Inc.

REFERENCES

- Ralph Alger Bagnold. 1941. *The physics of blown sand and desert dunes*. Methuen & Co, London.
- Bedrich Benes and Toney Roa. 2004. “Simulating desert scenery.”

- Luca Bruno, Nicolas Coste, Davide Fransos, Andrea Lo Giudice, Luigi Preziosi, and Lorenzo Raffaele. 2018. "Shield for sand: an innovative barrier for windblown sand mitigation." *Recent Patents on Engineering*, 12, 3, 237–246.
- Luca Bruno and Davide Fransos. 2015. "Sand transverse dune aerodynamics: 3D coherent flow structures from a computational study." *Journal of Wind Engineering and Industrial Aerodynamics*, 147, 291–301.
- Guillaume Cordonnier, Pierre Ecornier, Eric Galin, James Gain, Bedrich Benes, and Marie-Paule Cani. May 2018. "Interactive Generation of Time-evolving, Snow-Covered Landscapes with Avalanches." *Computer Graphics Forum*, 37, 2, (May 2018), 497–509.
- Sylvain Courrech du Pont. 2015. "Dune morphodynamics." *Comptes Rendus Physique*, 16, 1, 118–138.
- Bryan E. Feldman and James F. O'Brien. 2002. "Modeling the Accumulation of Wind-Driven Snow." In: *ACM SIGGRAPH Conference Abstracts and Applications*.
- M. Sturm G. E. Liston. 1998. "A snow-transport model for complex terrain." *Journal of Glaciology*, 44, 148.
- P. Rauwoens G. Strypsteen L.C. Van Rijn. July 2021. "Comparison of equilibrium sand transport rate model predictions with an extended dataset of field experiments at dry beaches with long fetch distance." *Aeolian Research*, 52, (July 2021).
- Kelly Kochanski, Gian-Carlo Defazio, Eric Green, Richard Barnes, Carlos Downie, Adam Rubin, and Barry Rountree. 2019. "Rescal-snow: Simulating snow dunes with cellular automata." *Journal of Open Source Software*, 4, 42, 1699.
- Jasper F Kok, Eric JR Parteli, Timothy I Michaels, and Diana Bou Karam. 2012. "The physics of wind-blown sand and dust." *Reports on progress in Physics*, 75, 10, 106901.
- L. Bruno L. Raffaele. 2019. "Windblown sand action on civil structures: Definition and probabilistic modelling." *Engineering Structures*, 148, 88–101.
- A Lo Giudice, G Giammanco, D Fransos, and L Preziosi. 2019. "Modeling sand slides by a mechanics-based degenerate parabolic equation." *Mathematics and Mechanics of Solids*, 24, 8, 2558–2575.
- Andrea Lo Giudice, Roberto Nuca, Luigi Preziosi, and Nicolas Coste. 2019. "Wind-blown particulate transport: A review of computational fluid dynamics models." *Mathematics in Engineering*, 1, 3, 508–547.
- Andrea Lo Giudice and Luigi Preziosi. 2020. "A fully Eulerian multiphase model of windblown sand coupled with morphodynamic evolution: Erosion, transport, deposition, and avalanching." *Applied Mathematical Modelling*, 79, 68–84.
- Clément Narteau, Deguo Zhang, Olivier Rozier, and Philippe Claudin. 2009. "Setting the length and time scales of a cellular automaton dune model from the analysis of superimposed bed forms." *Journal of Geophysical Research: Earth Surface*, 114, F3.
- Yen Ting Ng, Chohong Min, and Frédéric Gibou. 2009. "An efficient fluid–solid coupling algorithm for single-phase flows." *Journal of Computational Physics*, 228, 23, 8807–8829.
- G. Nichols. 2009. *Sedimentology and Stratigraphy*. A John Wiley & Sons, Ltd.
- C. Narteau O. Rozier. 2013. "A real-space cellular automaton laboratory." *Earth Surface Processes And Landforms*.
- K. Onoue and T. Nishita. 2000. "A method for modeling and rendering dunes with wind-ripples." In: *Proc. Pacific Conference on Computer Graphics and Applications*.
- Axel Paris, Adrien Peytavie, Eric Guérin, Oscar Argudo, and Eric Galin. 2019. "Desertscape simulation." In: *Computer Graphics Forum 7*. Vol. 38. Wiley Online Library, 47–55.
- Daan W Poppema, Andreas CW Baas, Suzanne JMH Hulscher, and Kathelijne M Wijnberg. 2022. "Cellular automaton modelling of the effects of buildings on aeolian bedform dynamics." *Aeolian Research*, 59, 100840.
- Daan W Poppema, Kathelijne M Wijnberg, Jan PM Mulder, and Suzanne JMH Hulscher. 2022. "Deposition patterns around buildings at the beach: Effects of building spacing and orientation." *Geomorphology*, 401, 108114.
- Lorenzo Raffaele, Nicolas Coste, and Gertjan Glabeke. 2022. "Life-cycle performance and cost analysis of sand mitigation measures: Toward a hybrid experimental-computational approach." *Journal of Structural Engineering*, 148, 7, 04022082.
- Nicolas Rosset, Guillaume Cordonnier, Regis Duvigneau, and Adrien Bousseau. 2023. "Interactive design of 2D car profiles with aerodynamic feedback." *Computer Graphics Forum*, 42, 2. <http://www-sop.inria.fr/revs/Basilic/2023/RCDB23>.
- Simon Schneiderbauer and Alexander Prokop. 2011. "The atmospheric snow-transport model: SnowDrift3D." *Journal of Glaciology*, 57, 203, 526–542.
- Simon Schneiderbauer, Thomas Tschachler, Johann Fischbacher, Walter Hinterberger, and Peter Fischer. 2008. "Computational fluid dynamic (CFD) simulation of snowdrift in alpine environments, including a local weather model, for operational avalanche warning." *Annals of glaciology*, 48, 150–158.
- Jos Stam. 1999. "Stable Fluids." In: *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '99)*. ACM Press/Addison-Wesley Publishing Co., USA, 121–128. ISBN: 0201485605.
- Brennen Taylor and John Keyser. 2023. "Real-Time Sand Dune Simulation." *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, 6, 1.
- Yoshihide Tominaga. 2018. "Computational fluid dynamics simulation of snowdrift around buildings: Past achievements and future perspectives." *Cold Regions Science and Technology*, 150, 2–14.

- V Vionnet, E Martin, V Masson, G Guyomarc'h, F Naaim-Bouvet, A Prokop, Y Durand, and C Lac. 2014. "Simulation of wind-induced snow transport and sublimation in alpine terrain using a fully coupled snowpack/atmosphere model." *The Cryosphere*, 8, 2, 395–415.
- Ning Wang and Bao-Gang Hu. 2012. "Real-Time Simulation of Aeolian Sand Movement and Sand Ripple Evolution: A Method Based on the Physics of Blown Sand." *Journal of Computer Science and Technology*, 27.
- BT Werner. 1995. "Eolian dunes: computer simulations and attractor interpretation." *Geology*, 23, 12, 1107–1110.
- Deguo Zhang, Clément Narteau, Olivier Rozier, and Sylvain Courrech du Pont. 2012. "Morphology and dynamics of star dunes from numerical modelling." *Nature Geoscience*, 5, 7, 463–467.
- Xuanyi Zhou and Tiange Zhang. 2023. "A review of computational fluid dynamics simulations of wind-induced snow drifting around obstacles." *Journal of Wind Engineering and Industrial Aerodynamics*, 234, 105350.
- AW Zingg. 1952. "Wind tunnel studies of the movement of sedimentary material." In: *Proc. 5th Hydraulics Conf., LAHR*, 111–135.