

# Przegląd 4

## Aerodynamika, algorytmy genetyczne, duże kroki i dynamika pozycji



Modelowanie fizyczne w animacji komputerowej  
Maciej Matyka

# Wykład z Modelowania – przegląd 4

1. Animation Aerodynamics

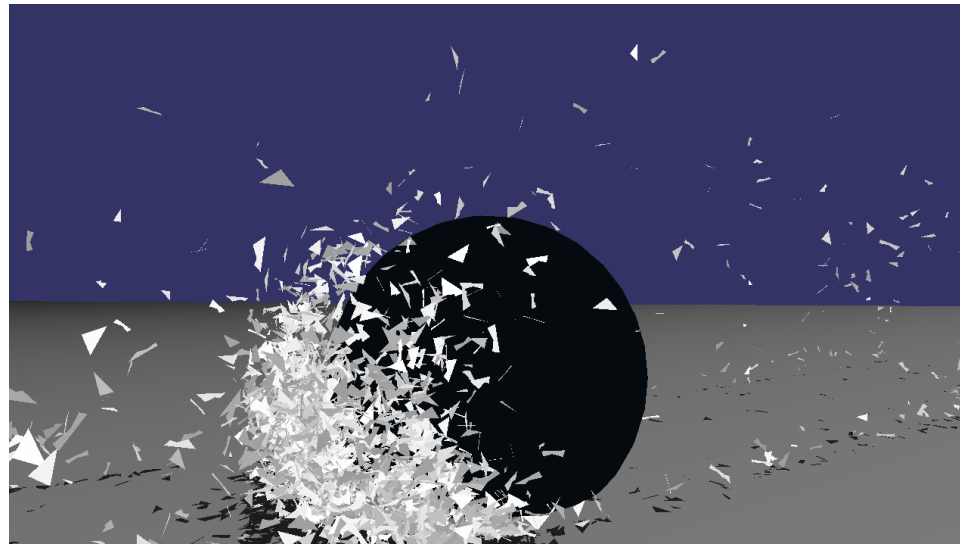
2. Algorytmy Genetyczne

3. Duże kroki w symulacji tkanin

4. Dynamika oparta na pozycji  
(nie siłach)

# Animation Aerodynamics

---



$$\rho \left( \underbrace{\frac{\partial \mathbf{u}}{\partial t}}_{\text{Unsteady acceleration}} + \underbrace{\mathbf{u} \cdot \nabla \mathbf{u}}_{\text{Convective acceleration}} \right) = \underbrace{-\nabla p}_{\text{Pressure gradient}} + \underbrace{\mu \nabla^2 \mathbf{u}}_{\text{Viscosity}} + \underbrace{\mathbf{f}}_{\text{Other forces}}$$

Inertia

$$\nabla \cdot \mathbf{u} = 0$$

Continuity equation

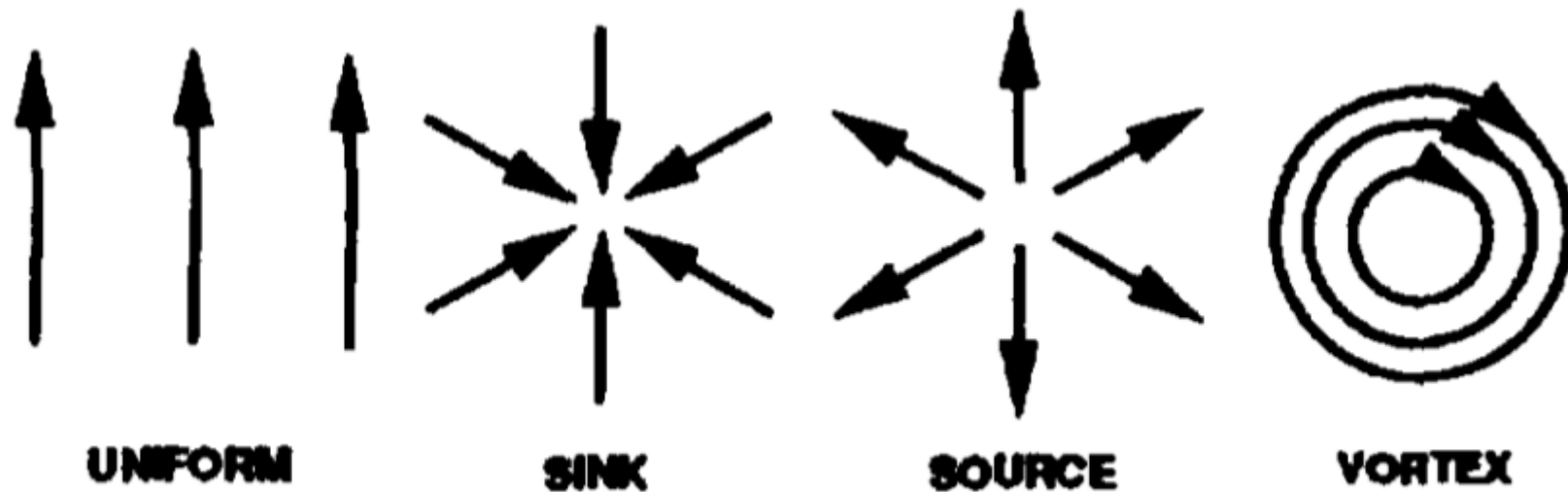
# Równania Naviera-Stokesa

$$\rho \left( \underbrace{\frac{\partial \mathbf{u}}{\partial t}}_{\text{Unsteady acceleration}} + \underbrace{\mathbf{u} \cdot \nabla \mathbf{u}}_{\text{Convective acceleration}} \right) = \underbrace{-\nabla p}_{\text{Pressure gradient}} + \underbrace{\mu \nabla^2 \mathbf{u}}_{\text{Viscosity}} + \underbrace{\mathbf{f}}_{\text{Other forces}}$$

$\nabla \cdot \mathbf{u} = 0$   
 Continuity equation

Tu mają rozwiązanie analityczne!

*Animation Aerodynamics, 1991*  
Wejchert, J. et. al.



**Figure 1.** A schematic diagram of the flow primitives.

# Prymitywy we współrzędnych cylindrycznych

Using cylindrical coordinates, the potential and the velocity field for a line of source at the origin, with strength  $a$ , is:

$$\phi = \frac{a}{2\pi} \ln r; \quad v_r = \frac{a}{2\pi r}; \quad v_\theta = 0; \quad v_z = 0. \quad (2)$$

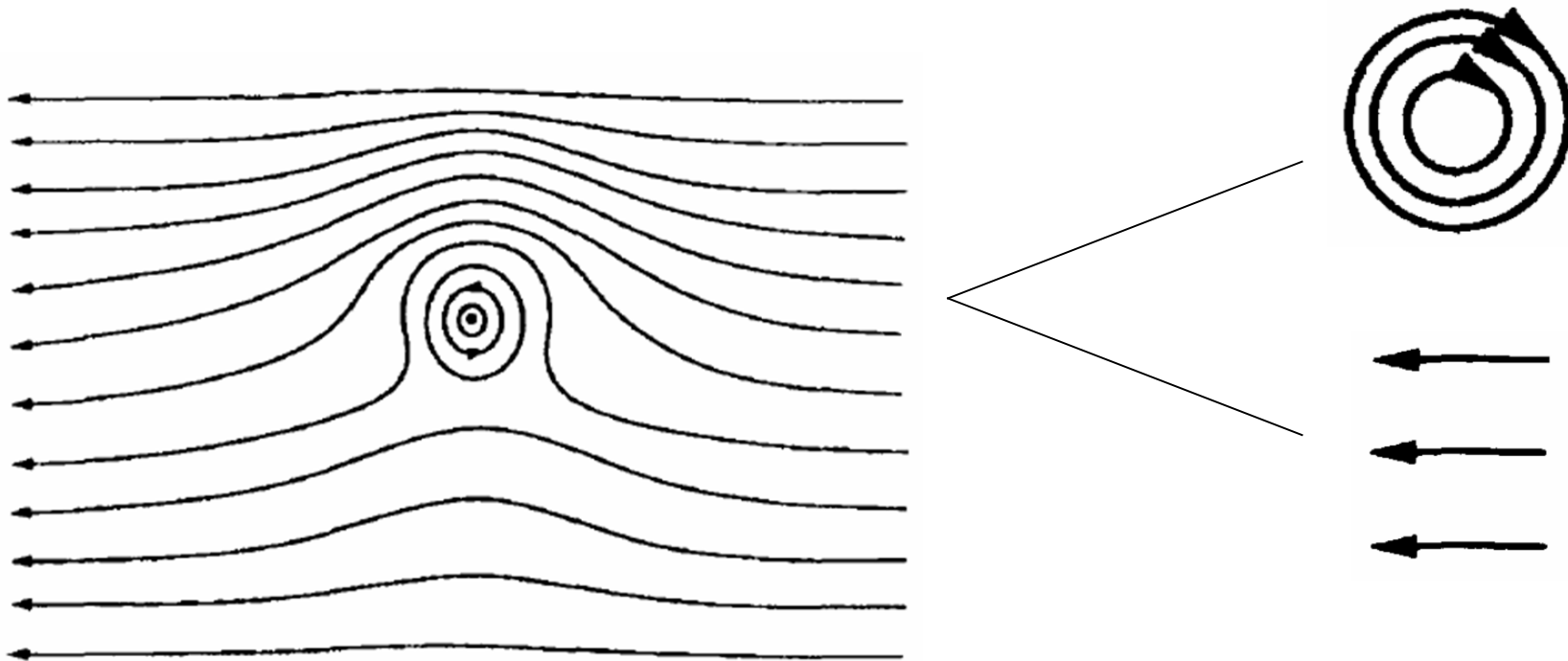
For a sink the constant  $a$  is set negative. A vortex at the origin with strength  $b$  is given by:

$$\phi = \frac{b}{2\pi} \theta; \quad v_r = 0; \quad v_\theta = \frac{b}{2\pi r}; \quad v_z = 0. \quad (3)$$

# *Animation Aerodynamics, 1991*

Wejchert, J. et. al.

$$\mathbf{V} = \mathbf{v}_{\text{vort}}(x, y, z) + \mathbf{v}_{\text{sink}}(x, y, z) + \mathbf{v}_{\text{source}}(x, y, z) + \dots$$



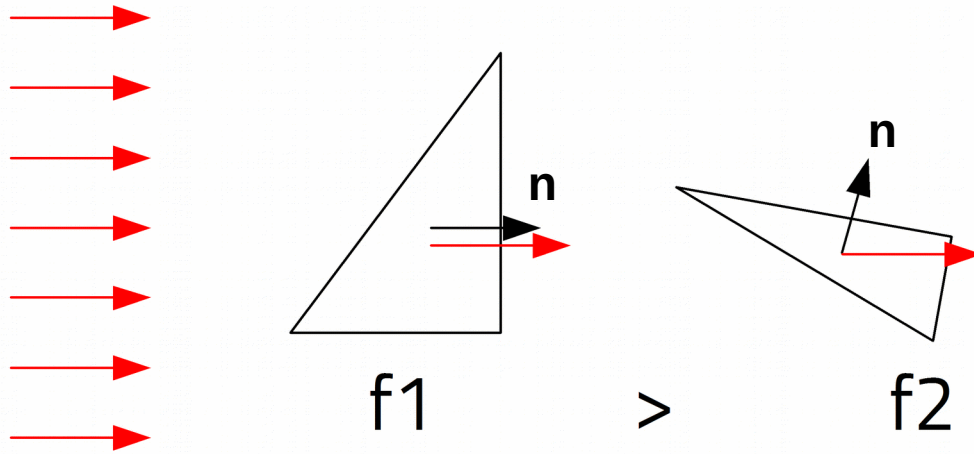


# Sila Stokes'a

$$\mathbf{F} = 6\pi a\eta\mathbf{v}'$$

$$\mathbf{v}' = \mathbf{v} - \mathbf{p}$$

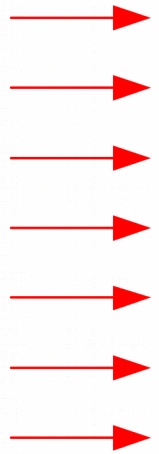
# Siła wiatru



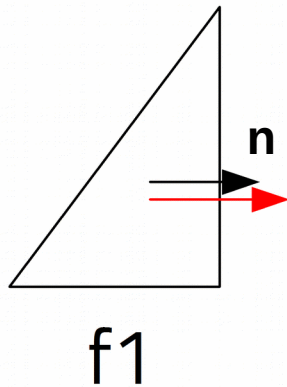
wind  $u$

(można też dodać siłę pochodzącą od względnej różnicy prędkości - tarcie..)

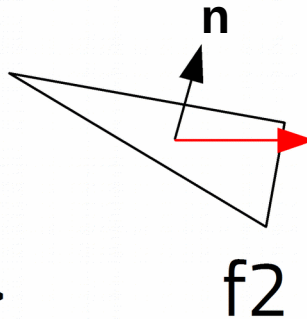
# Siła wiatru



wind  $u$



>



$$\mathbf{F} = \mathbf{n} * (\mathbf{n} * \mathbf{u})$$

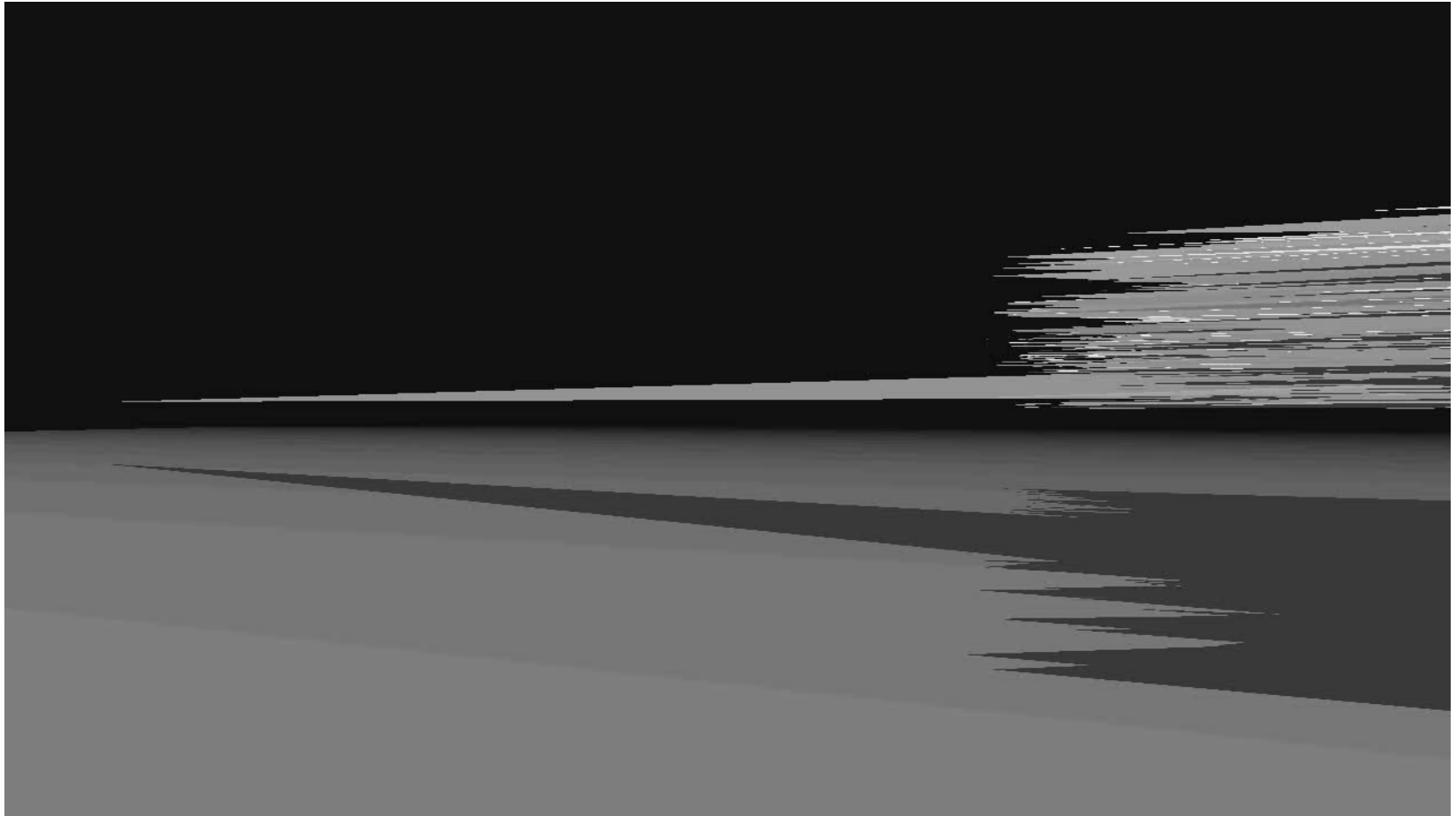
Iloczyn skalarny

(można też dodać siłę pochodzącą od względnej różnicy prędkości - tarcie..)

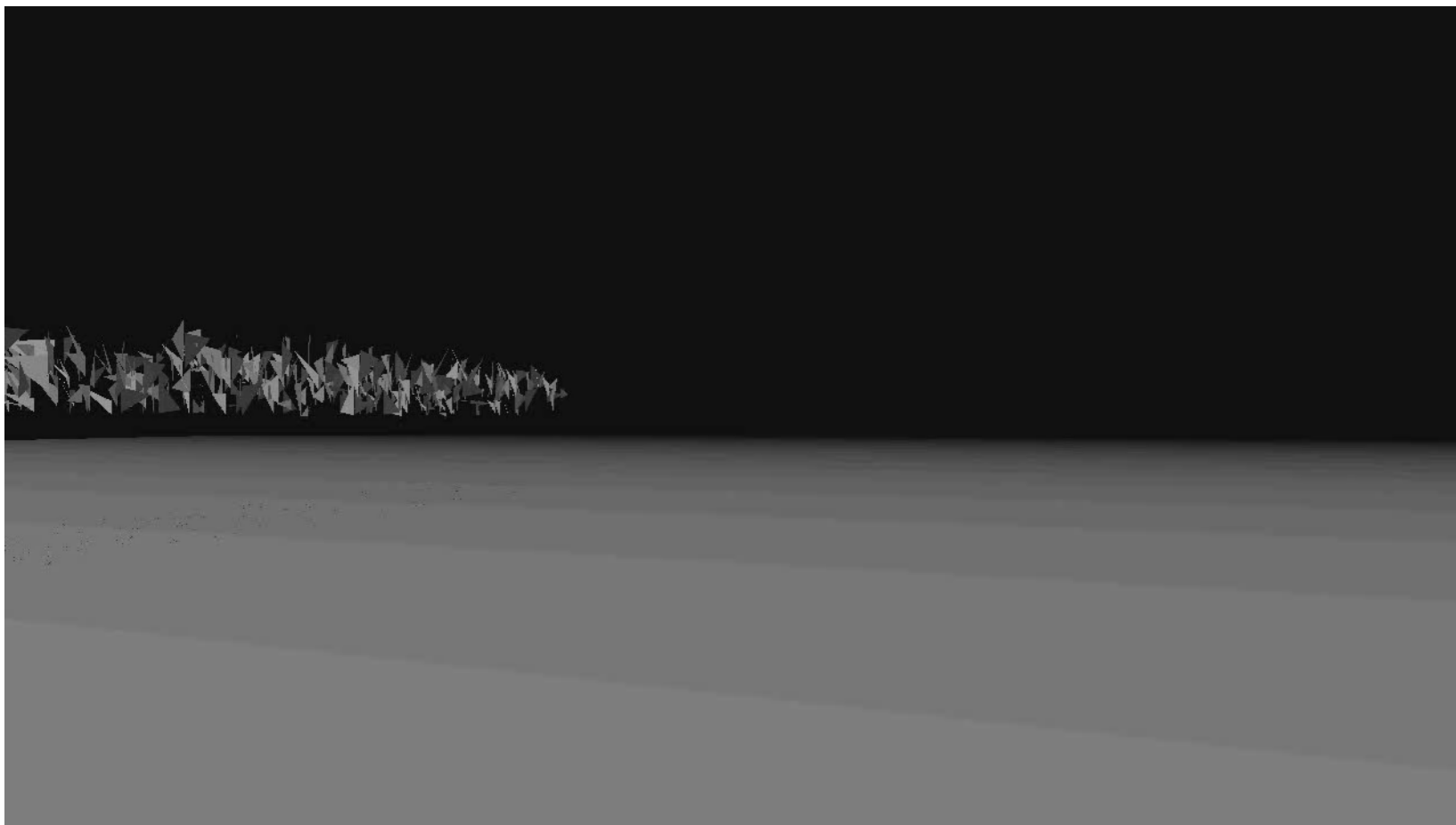
# Siły dla każdego trójkąta

$$\mathbf{F} = \mathbf{F}_{grav} + \mathbf{F}_{spring} + \mathbf{F}_{fluid} + \dots$$

# Czasem zdarza się i tak...



leaves1-fail1.wmv



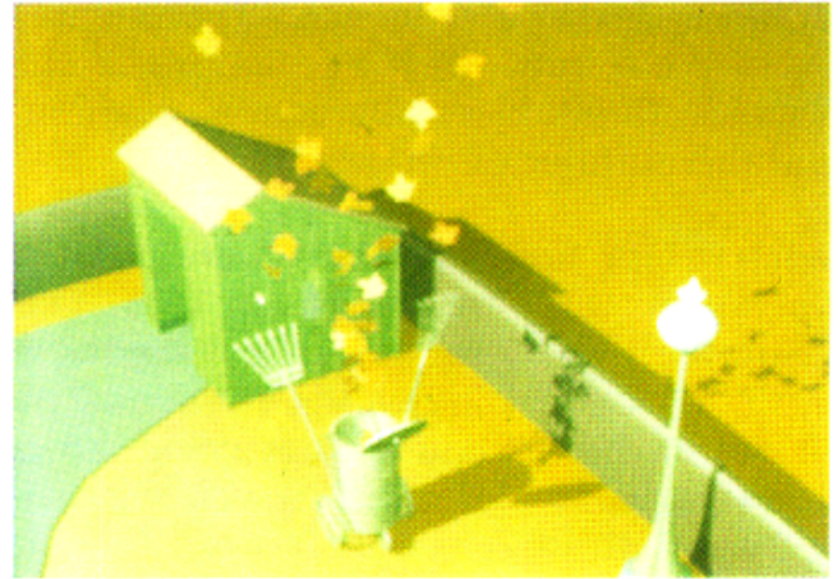
**Podobny model w działaniu**



leaves2.wmv



**Figure 8.** Leaves being blow up and over a slide.



**Figure 7.** Leaves being sucked in by a garbage bin.

- J. Wejchert, D. Haumann,  
Animation Aerodynamics  
Computer Graphics, Volume 25, Number 4, July 1991

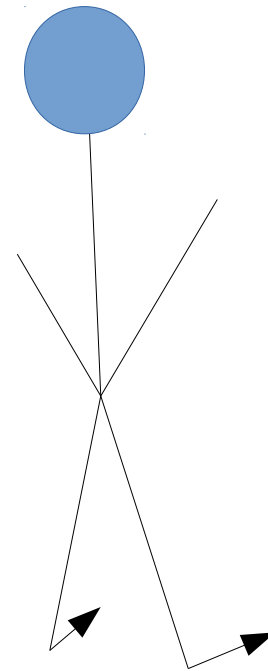
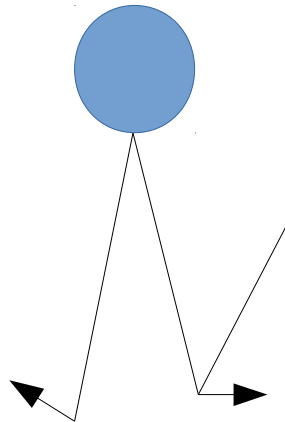
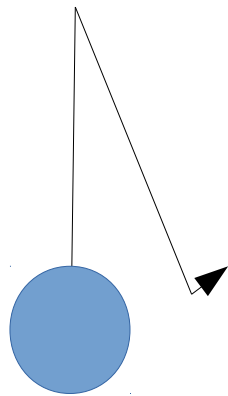
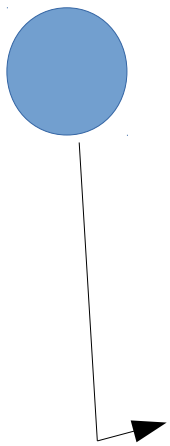
# Algorytmy Genetyczne

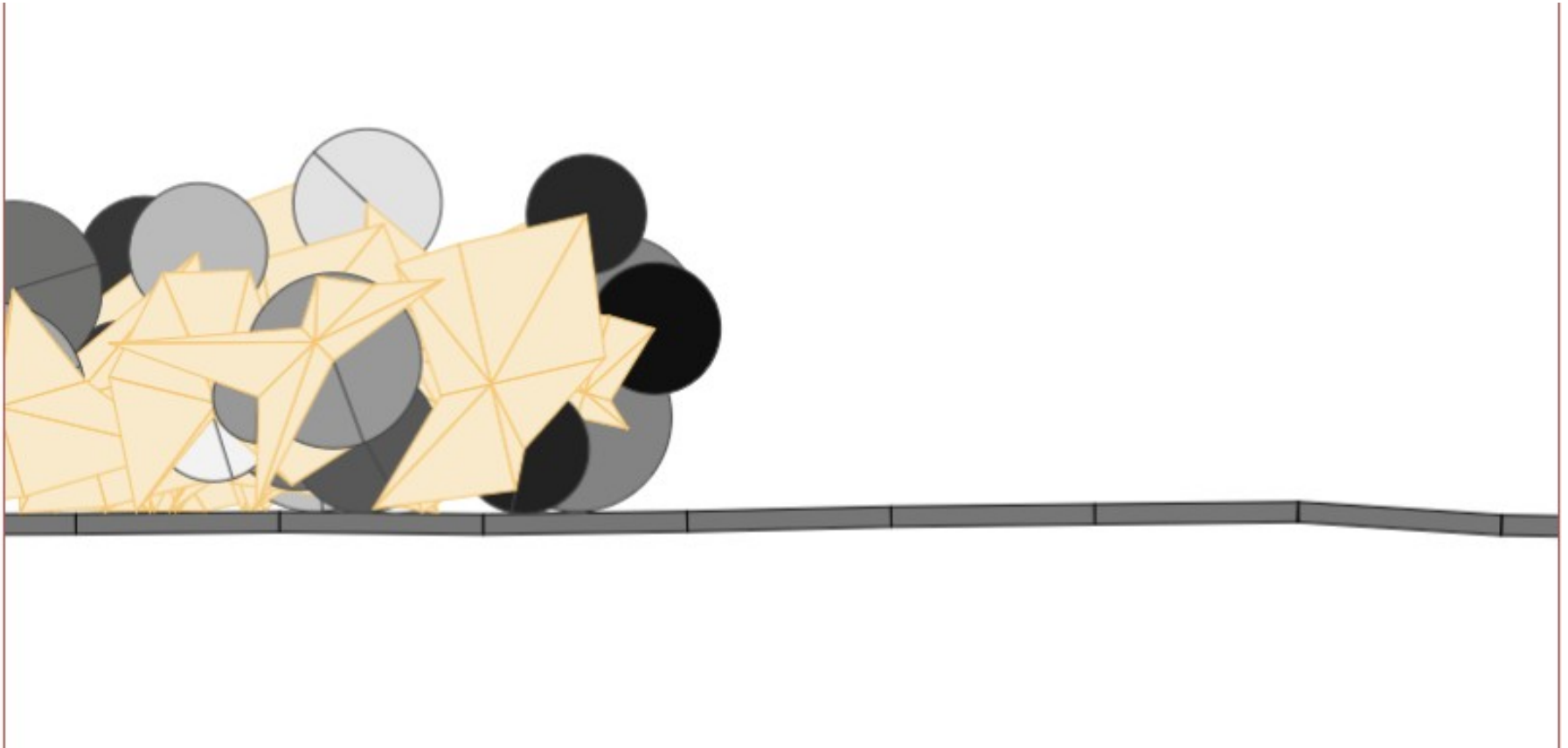


- Genetic Cars (WWW)
- Medium.com article



- Który z obiektów lepiej będzie chodził?





[http://rednuht.org/genetic\\_cars\\_2/](http://rednuht.org/genetic_cars_2/)

# Algorytm genetyczny [\[edytuj\]](#)

---

**Algorytm genetyczny** – rodzaj [heurystyki](#) przeszukującej przestrzeń alternatywnych rozwiązań problemu w celu wyszukania rozwiązań najlepszych.

[https://pl.wikipedia.org/wiki/Algorytm\\_genetyczny](https://pl.wikipedia.org/wiki/Algorytm_genetyczny)

# Rozwiązanie problemu z użyciem algorytmów genetycznych wymaga:

- **zakodowania cech naszego układu** (obiektu) w postaci łańcucha znaków lub symboli (genom) tak, aby z niego można było wygenerować rozwiązanie (osobnik)
- rozwiązanie może być wtedy **mutowane** przez zmianę symboli w tym łańcuchu znaków (genomie) w sposób losowy
- **pomiar tzw. fitness**, czyli miary określającej jak blisko jesteśmy naszego rozwiązania (osobnika) poprawnego

# Algorytm

- **Algorytm genetyczny** polega na **minimalizacji miary fitness** z użyciem mutacji genomu naszego osobnika.
- Osobnikiem może być zwierzę, człowiek, wirus —to w biologii, ale też... rysunek, animacja, obiekt fizyczny (i to interesuje nas tutaj).

# Algorytm c.d.

1. Wygeneruj **genom pierwszego osobnika**—matki
2. Zmierz **fitness** matki
3. Wykonaj **losową mutację** genomu
4. Wygeneruj **nowego osobnika** (dziecko)
5. Zmierz **fitness dziecka**
6. Jeśli fitness dziecka jest większy od fitnessu matki —przepisz genom dziecka do genomu matki
7. Idź do kroku 3.

# Kompresja rysunku (Algorytm Genetyczny)

- Jak przybliżyć rysunek za pomocą zbioru przeźroczystych figur (tu kółek)?
- Genom koduje:
  - Pozycję  $N$  kół
  - Ich kolory R,G,B
  - Stopień przeźroczystości
- Osobnik: rysunek (zbiór wartości R,G,B)
- Fitness – stopień podobieństwa do Oryginału

# 265 mutaciji

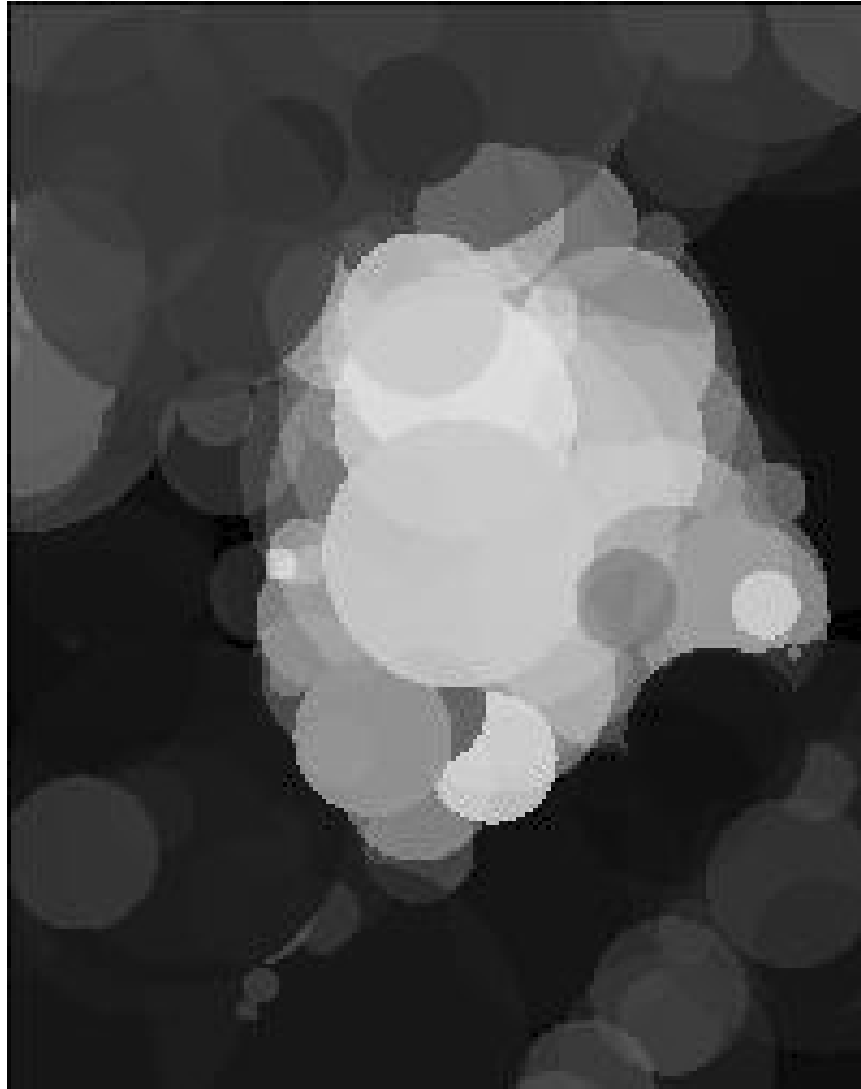




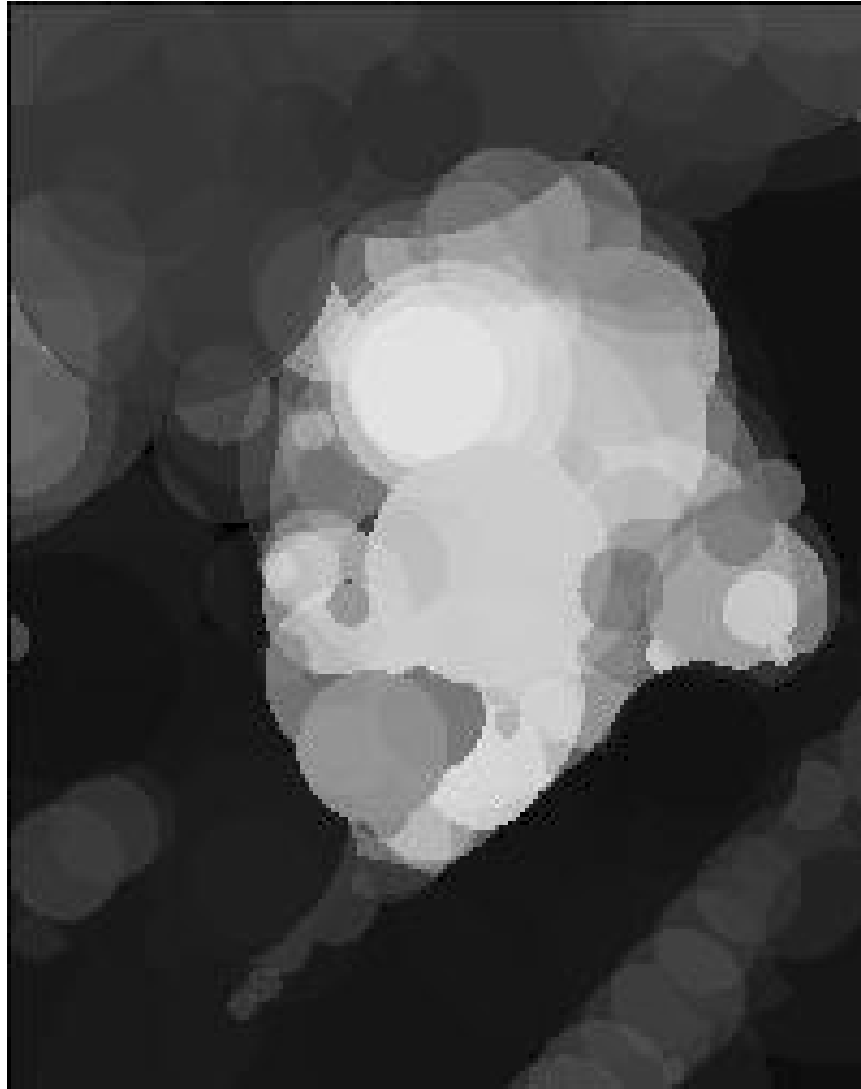
2203 mutacije



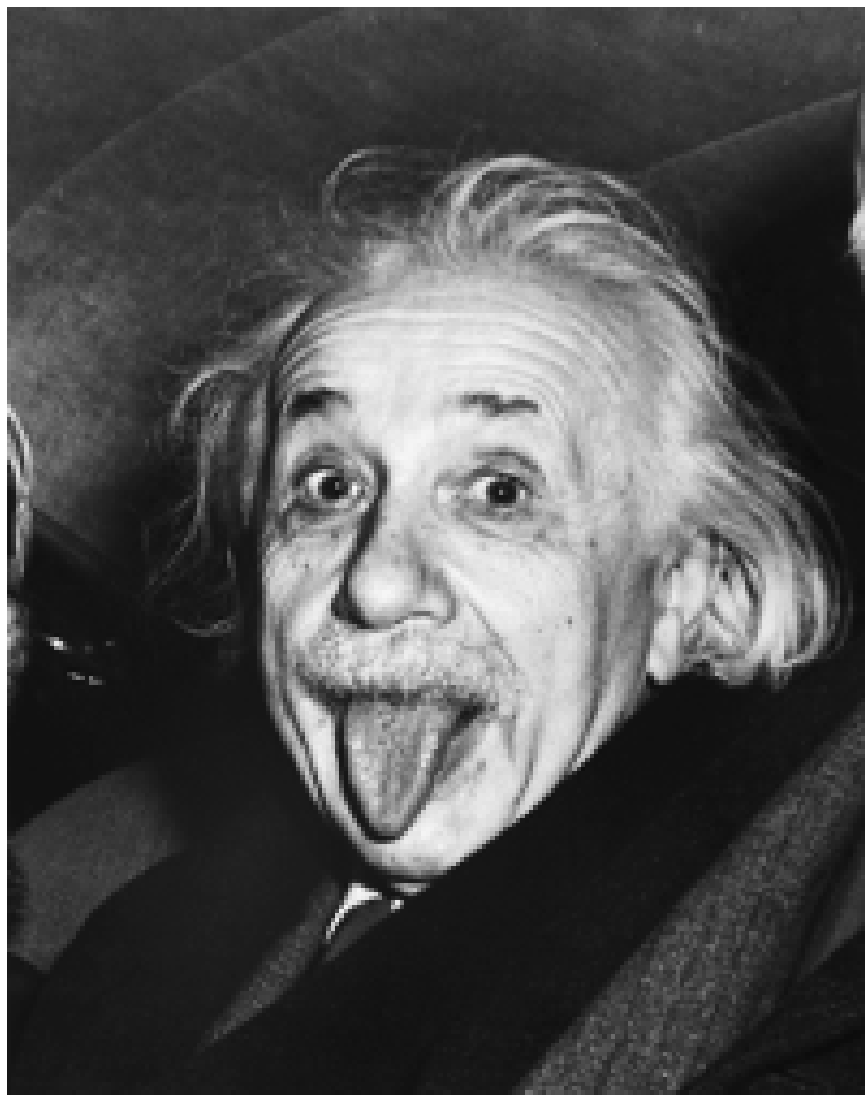
10119 mutaciji



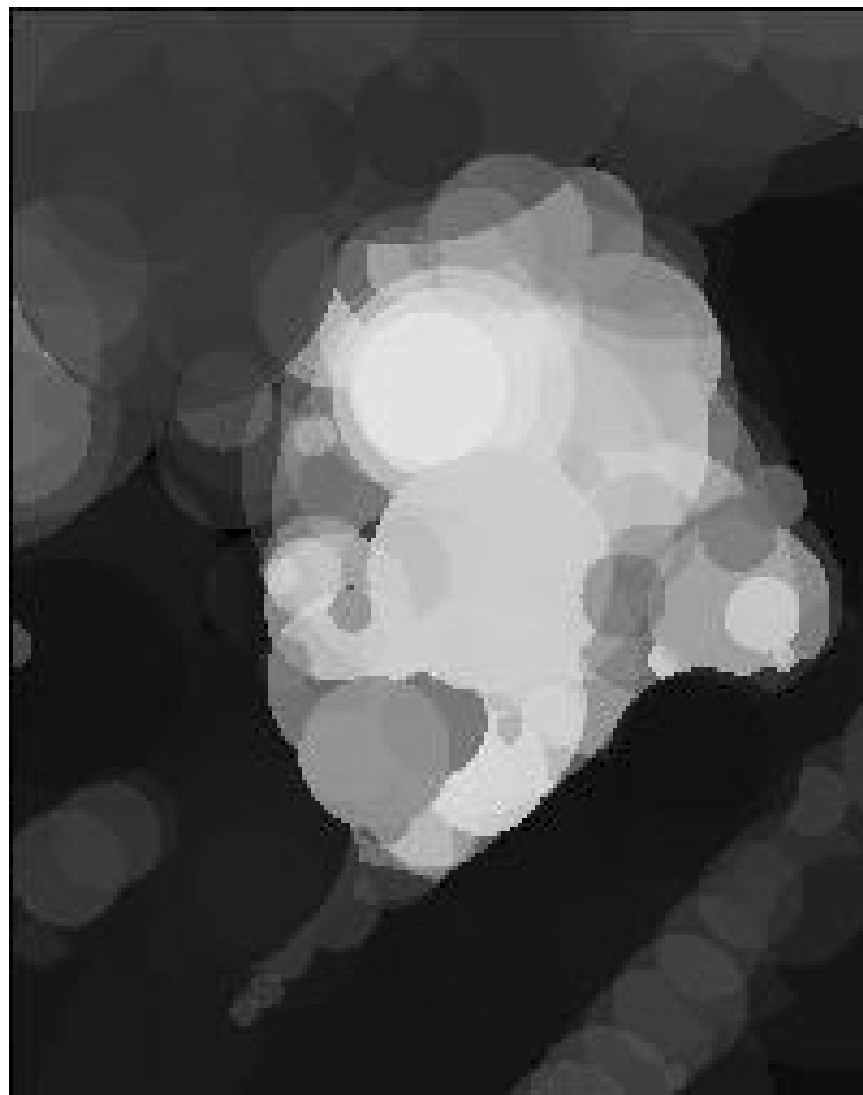
4832304 mutacjii



Oryginał

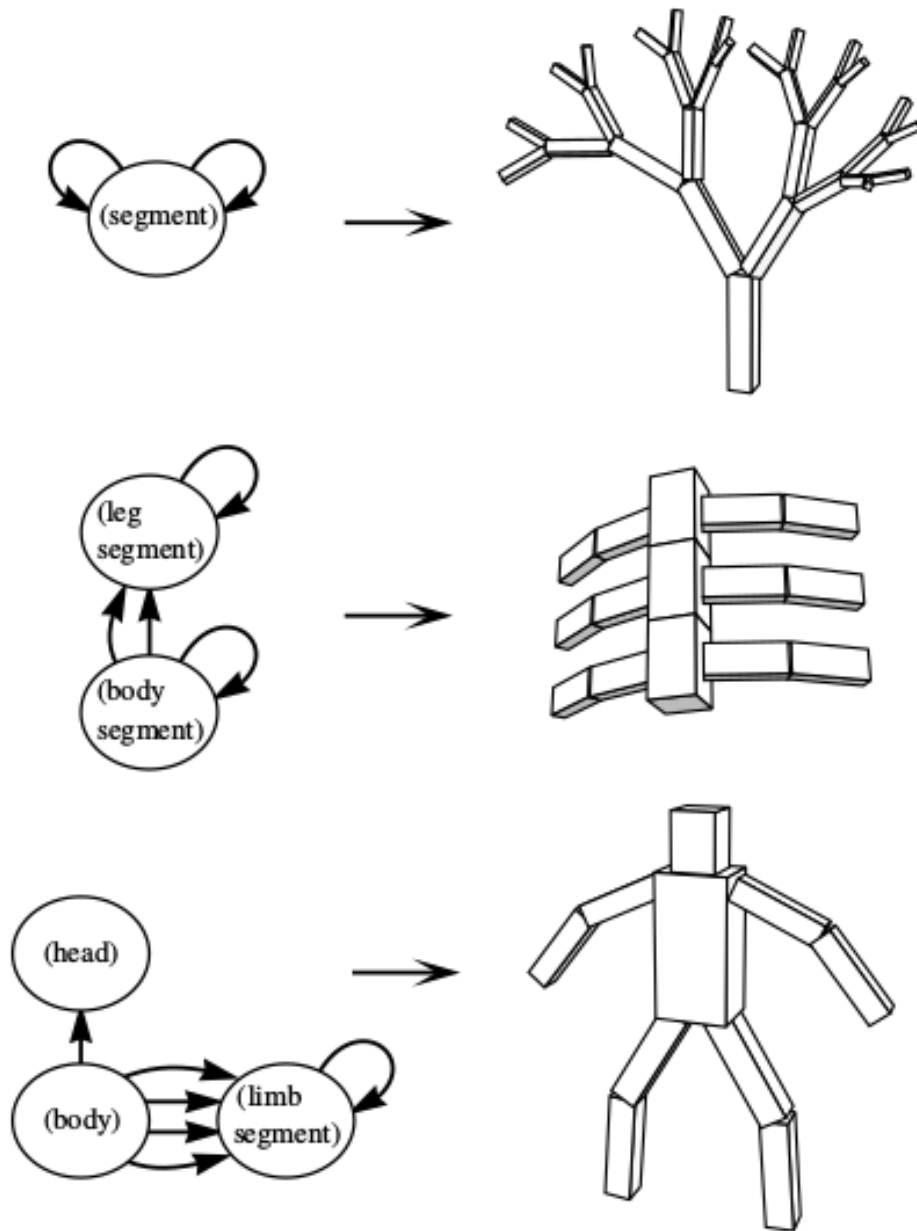


4832304 mutacji



**Genotype:** directed graph.

**Phenotype:** hierarchy of 3D parts.



## Evolving Virtual Creatures

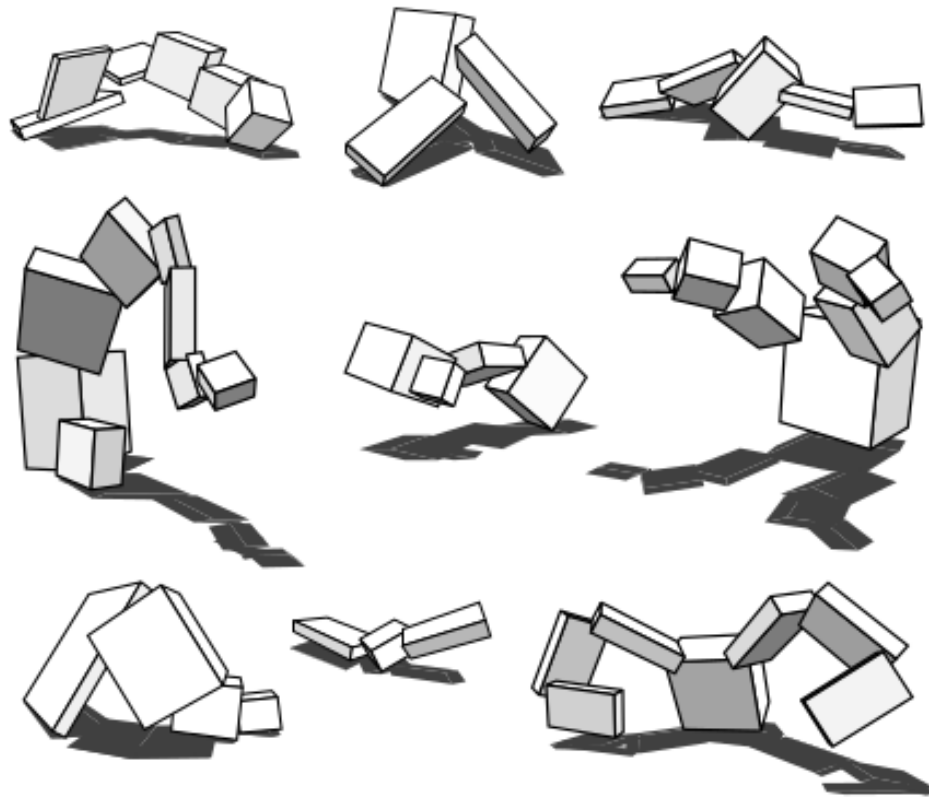
Karl Sims

Thinking Machines Corporation  
245 First Street, Cambridge, MA 02142

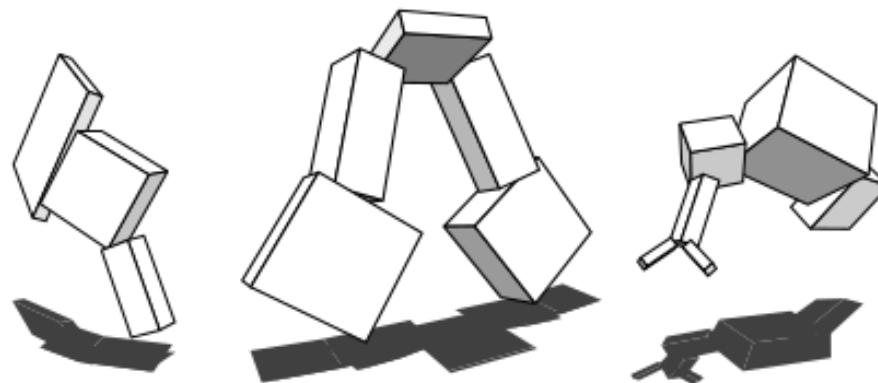
+ sensory  
+ mutacje / ewolucja

**Figure 1:** Designed examples of genotype graphs and corresponding creature morphologies.

# Wyniki



**Figure 7:** Creatures evolved for walking.



**Figure 8:** Creatures evolved for jumping.

# Karl Sims

## Evolving 3D Morphology and Behavior by Competition

Karl Sims

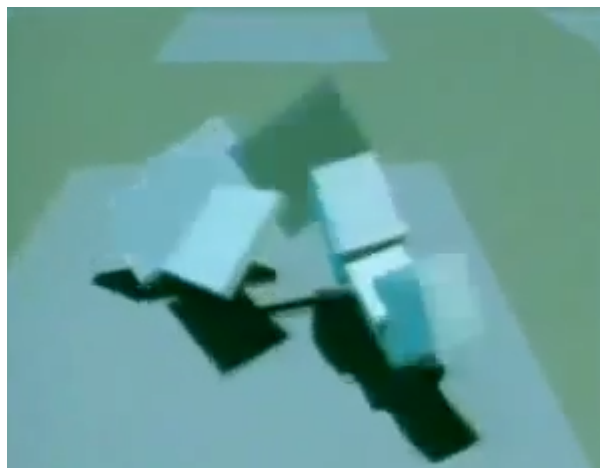
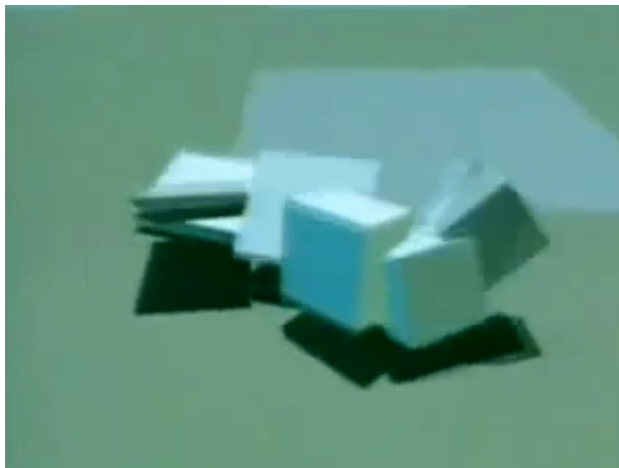
Thinking Machines Corporation  
245 First Street, Cambridge, MA 02142

### Abstract

This paper describes a system for the evolution and co-evolution of virtual creatures that compete in physically simulated three-dimensional worlds. Pairs of individuals enter one-on-one contests in which they contend to gain control of a common resource. The winners receive higher relative fitness scores

The work presented here takes the former approach. The fitness of an individual is highly dependent on the specific behaviors of other individuals currently in the population. The hope is that virtual creatures with higher complexity and more interesting behavior will evolve than when applying the selection pressures of optimization alone.

Many simulations of co-evolving populations have been

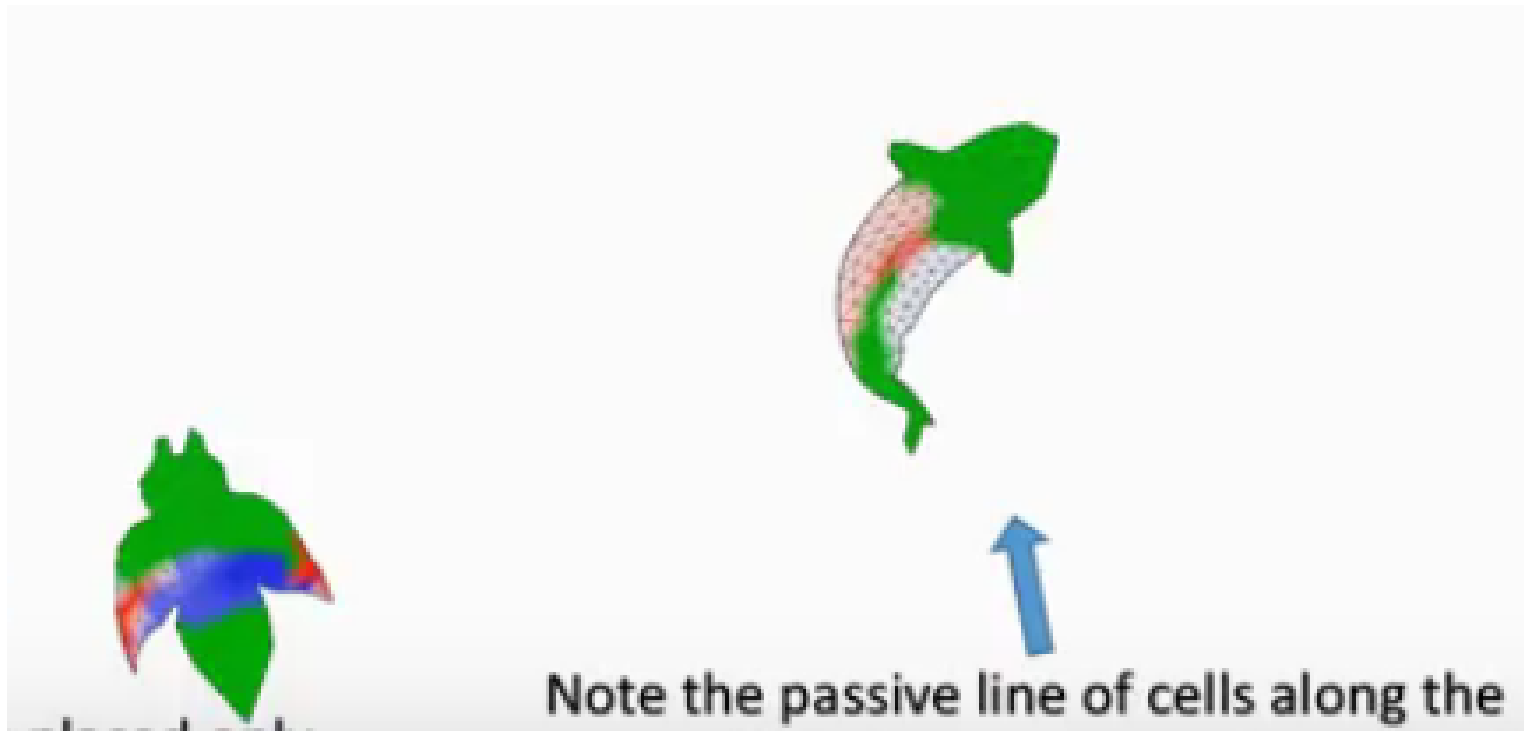


Karl Sims - Evolved  
Virtual Creatures,  
Evolution Simula...

Karl Sims - Evolved Virtual Creatures, Evolution Simulation, 1994 [360p].mp4

**Karl Sims - Evolved Virtual Creatures, Evolution Simulation, 1994**

# Genetyka + symulacje w akcji



<https://www.youtube.com/watch?v=CXTZHHQ7ZiQ>

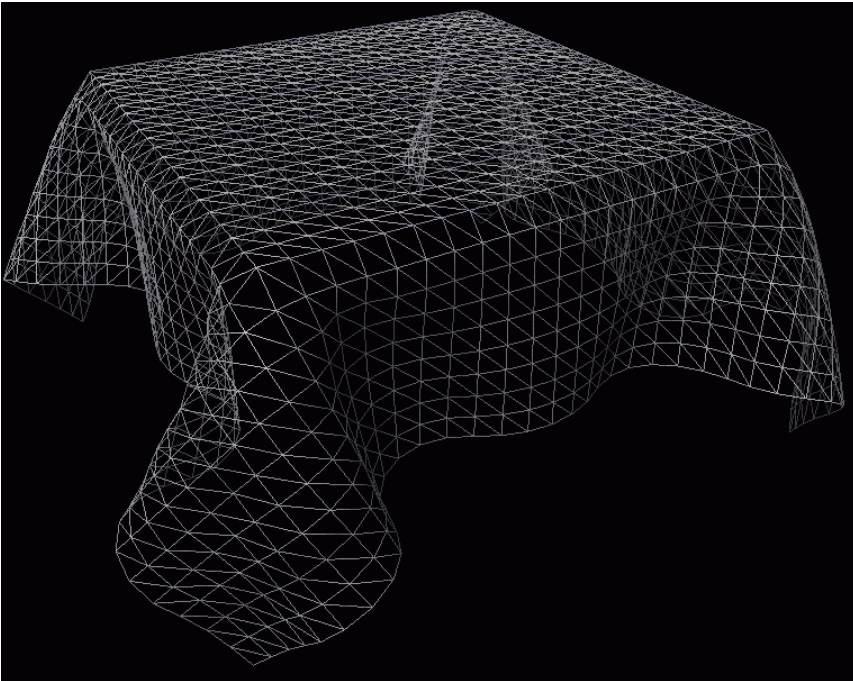
Harnessing evolutionary creativity\_ evolving soft-bodied animats in simulated physical environments [720p]



# Duże kroki w symulacji tkanin (Large Steps in Cloth Simulation)



# Budowa modeli tkanin



# Problemy z całkowaniem w Spring-Mass

- Krok czasowy musi być mały
- Też w przypadku Verleta i MidPoint
- Metody Euler/Verlet/MidPoint są jawne

# Large Steps in Cloth Simulation

David Baraff    Andrew Witkin

Robotics Institute  
Carnegie Mellon University

## Abstract

The bottle-neck in most cloth simulation systems is that time steps must be small to avoid numerical instability. This paper describes a cloth simulation system that can stably take large time steps. The simulation system couples a new technique for enforcing constraints on individual cloth particles with an implicit integration method. The simulator models cloth as a triangular mesh, with internal cloth forces derived using a simple continuum formulation that supports modeling operations such as local anisotropic stretch or compression; a unified treatment of damping forces is included as well. The implicit integration method generates a large, unbanded sparse linear system at each time step which is solved using a modified conjugate gradient method that simultaneously enforces particles' constraints. The constraints are always maintained exactly, independent of the number of conjugate gradient iterations, which is typically small. The resulting simulation system is significantly faster than previous accounts of cloth simulation systems in the literature.

**Keywords**—Cloth, simulation, constraints, implicit integration, physically-based modeling.

tion of  $\mathbf{x}$ —yields the cloth's internal energy, and  $\mathbf{F}$  (a function of  $\mathbf{x}$  and  $\dot{\mathbf{x}}$ ) describes other forces (air-drag, contact and constraint forces, internal damping, etc.) acting on the cloth.

In this paper, we describe a cloth simulation system that is much faster than previously reported simulation systems. Our system's faster performance begins with the choice of an *implicit* numerical integration method to solve equation (1). The reader should note that the use of implicit integration methods in cloth simulation is far from novel: initial work by Terzopoulos *et al.* [15, 16, 17] applied such methods to the problem.<sup>1</sup> Since this time though, research on cloth simulation has generally relied on *explicit* numerical integration (such as Euler's method or Runge-Kutta methods) to advance the simulation, or, in the case of energy minimization, analogous methods such as steepest-descent [3, 10].

This is unfortunate. Cloth strongly resists stretching motions while being comparatively permissive in allowing bending or shearing motions. This results in a “stiff” underlying differential equation of motion [12]. Explicit methods are ill-suited to solving stiff equations because they require many small steps to stably advance the simulation forward in time.<sup>2</sup> In practice, the computational cost of an explicit method greatly limits the realizable resolution of the cloth. For some applications, the required spatial resolution—that

# „Large Steps...” w pigułce

- Metoda niejawna całkowania równań ruchu
- Układ równań liniowych
- Macierz rzadka
- Metoda iteracyjna gradientów sprzężonych  
(Conjugate Gradient)
- Możemy użyć... dowolnie dużego kroku czasowego  $DT$



Large Steps in Cloth Simulation  
David Baraff  
Andrew Witkin

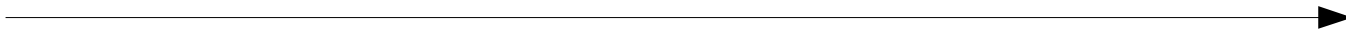
# Aplikacja

- Cloth + kolizje + całkowanie (?)



<https://www.youtube.com/watch?v=KBfxnaylIOY>

# Position Based Dynamics



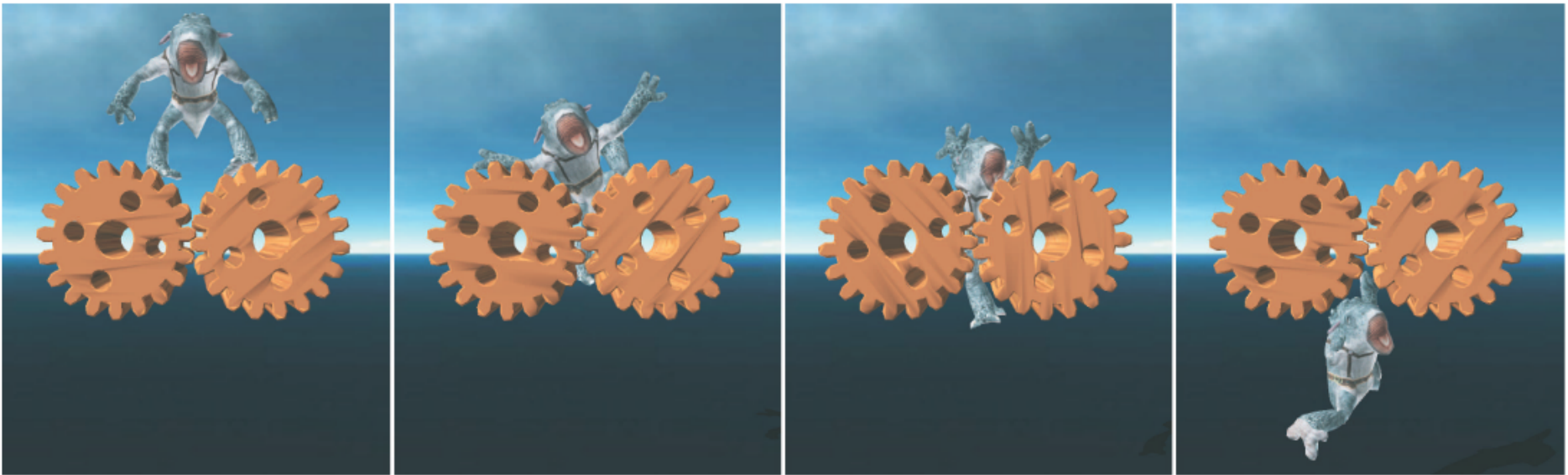


3<sup>rd</sup> Workshop in Virtual Reality Interactions and Physical Simulation "VRIPHYS" (2006)  
C. Mendoza, I. Navazo (Editors)

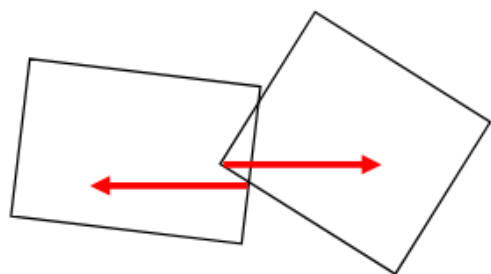
# Position Based Dynamics

Matthias Müller Bruno Heidelberger Marcus Hennix John Ratcliff

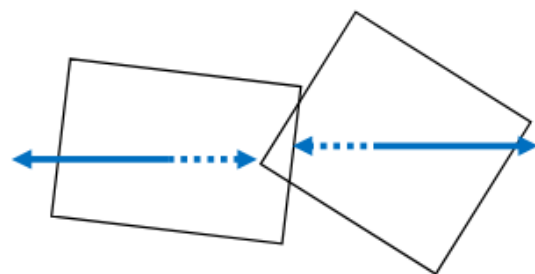
AGEIA



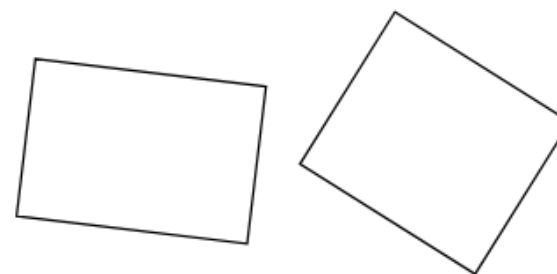
# Standardowy model z siłami



rozpoznana  
kolizja



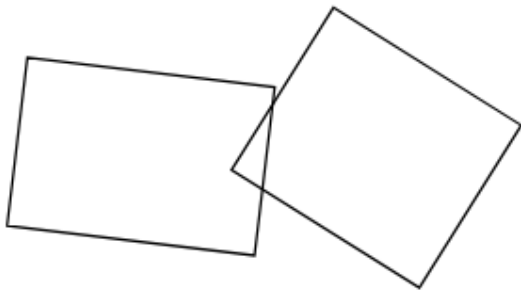
Wyliczamy siły →  
Przyspieszenia → prędkości



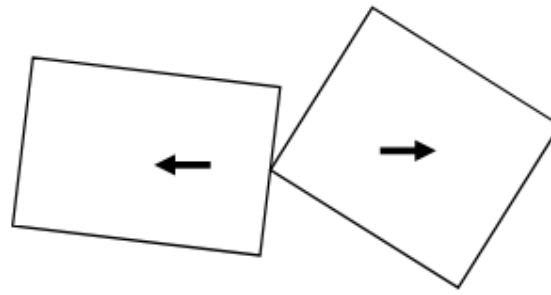
Przesuwamy ciała

- Opóźnienie w reakcji
- „Nie można ustawić dużego  $ks$ ” - czyli problemy z krokiem czasowym (oscylacje)

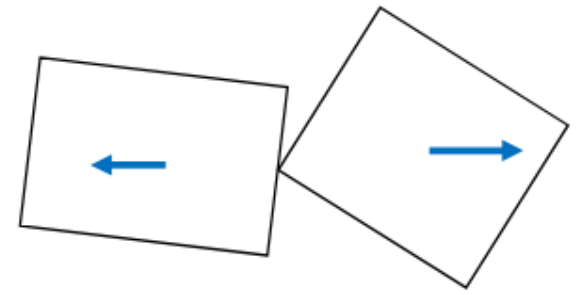
# Dynamika bez przyspieszeń



rozpoznana  
kolizja



przesuń tak, aby nie  
Kolidowały ze sobą



koryguj prędkości

# Algorytm symulacji PBD

Inicjalizuj  $x_0, v_0$

Pętla

$$v(n) \leftarrow v(n) + dt * f(x(n))$$

$$p \leftarrow x(n) + dt * v(n)$$

$$x(n+1) \leftarrow \text{modyfikuj } p \quad \text{korekta pozycji}$$

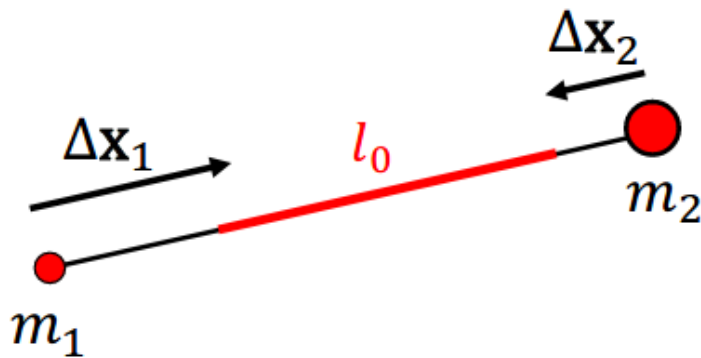
$$u \leftarrow (x(n+1) - x(n)) / dt \quad \text{wylicz prędkość}$$

$$v(n+1) \leftarrow u \quad \text{korekta prędkości}$$

Koniec pętli

# Sprężynki z PBD

- Wprowadzamy wagi dla więzów (tu  $w_1$  i  $w_2$ )

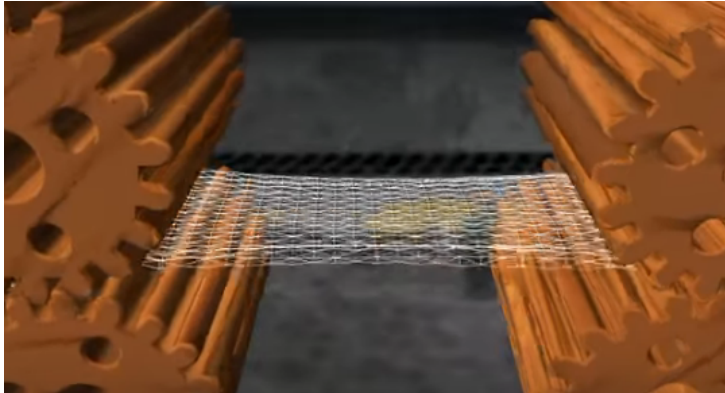
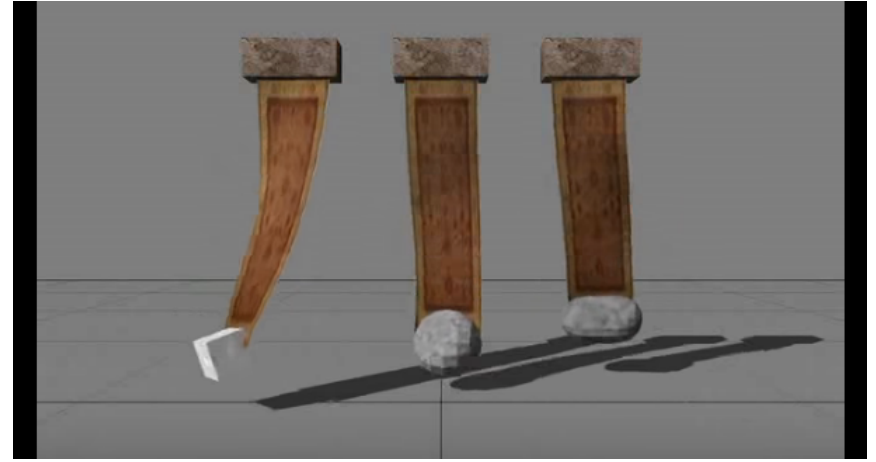
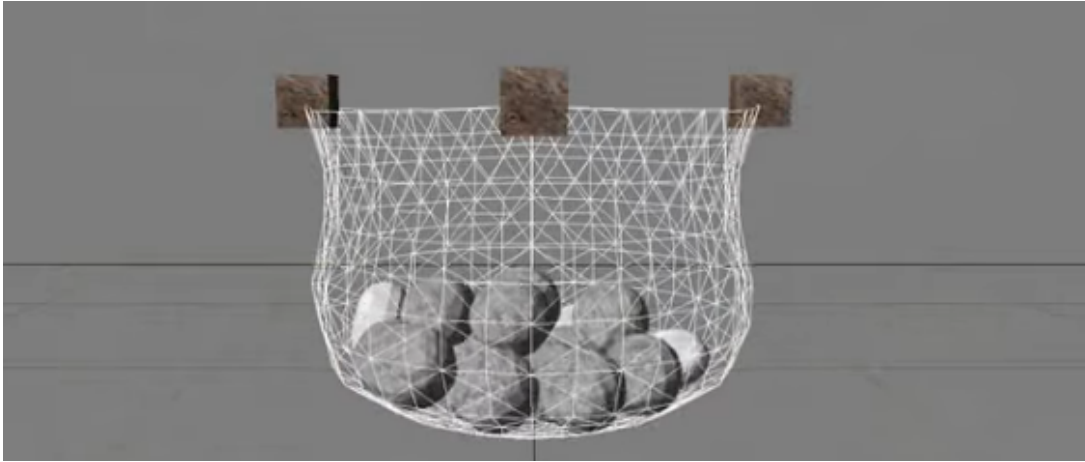


$$\Delta \mathbf{x}_1 = -\frac{w_1}{w_1 + w_2} (|\mathbf{x}_1 - \mathbf{x}_2| - l_0) \frac{\mathbf{x}_1 - \mathbf{x}_2}{|\mathbf{x}_1 - \mathbf{x}_2|}$$

$$\Delta \mathbf{x}_2 = +\frac{w_2}{w_1 + w_2} (|\mathbf{x}_1 - \mathbf{x}_2| - l_0) \frac{\mathbf{x}_1 - \mathbf{x}_2}{|\mathbf{x}_1 - \mathbf{x}_2|}$$

- Kolejne ograniczenia ruchu (siły) to będą więzy z różnymi wagami
- Iteracje Gaussa-Seidla  
(tego już nie będziemy omawiać na wykładzie)

# PBD w działaniu



Position Based Dynamics [360p].mp4

<https://youtu.be/j5igW5-h4ZM>

Koniec

