

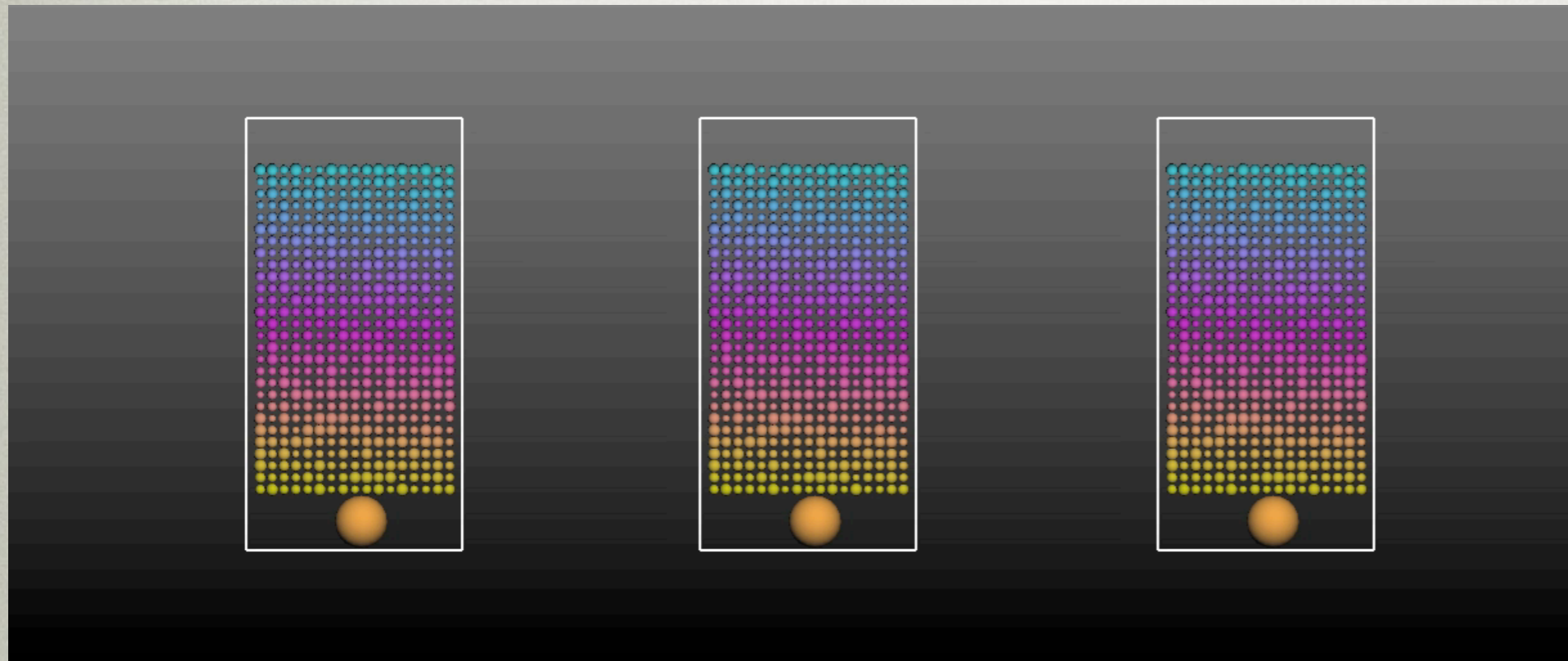
# EXERCISE 11

OPTIMIZATION AND BRAZILNUTS

# EXERCISE 11

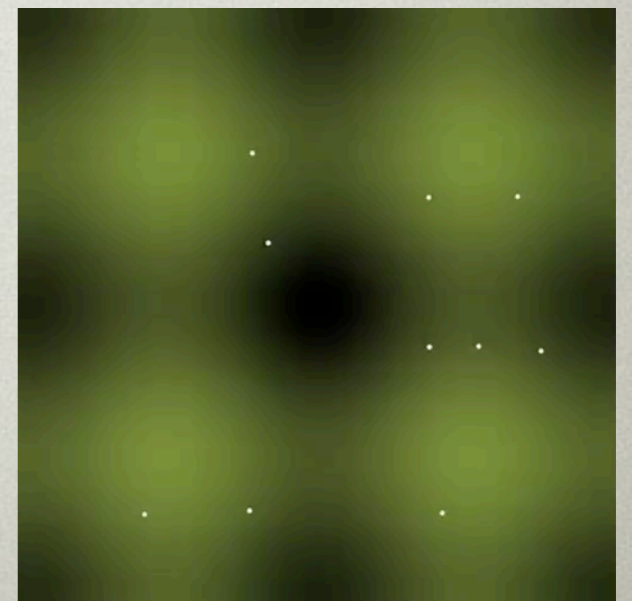
## OPTIMIZATION

---



Granular Flow: Brazil Nut Effect

Particle Swarm Optimization



# EXERCISE 11

## QUESTION 1

SwarmingParticle.h

Particle Swarm Optimization:

```
void updatePBest(const double f)
{
    //QUESTION 1:
    // check whether f is better (i.e. smaller) than the best f
    // seen so far (stored as fpbest) and update pbest and fpbest accordingly.
    //START FILLING HERE
    //
    //STOP FILLING HERE
}

//update my information copy about the global best point if f is better
void updateLBest(const double f, const double * const xin)
{
    if (f < flbest)
    {
        flbest = f;

        for (int i = 0; i < Dim; ++i) {
            lbest[i] = xin[i];
        }
    }
}

//update my position and let's hope to have the best overall score!
void update(const double * const rnd1, const double * const rnd2)
{
    //QUESTION 1:
    // update velocity and position of the particles
    // and make sure the particles' locations are in the feasible domain.
    //START FILLING HERE
    //
    //STOP FILLING HERE
}
```

Implement the tracking of the particles own best position

Implement the position and velocity update of the swarming particle

# EXERCISE 1 1

## QUESTION 2

---

### Optimize the Rising of the Nut

In order to find the optimal shake parameter that speed up the rising of the Brazil Nut, we link the problem to the Particle Swarm Optimizer introduced in Question 1

Start the Particle Swarm Optimizer with the argument *brazilnut*, indicating the optimization problem, the executable that evaluates the fitness function and the final time.

```
./pso brazilnut ./bn 30
```

When you are satisfied with the result of the optimization process, you can write down the best result together with the parameter set.

Launch the Brazil Nut simulation with the best set of parameters, saving the images and make a movie.