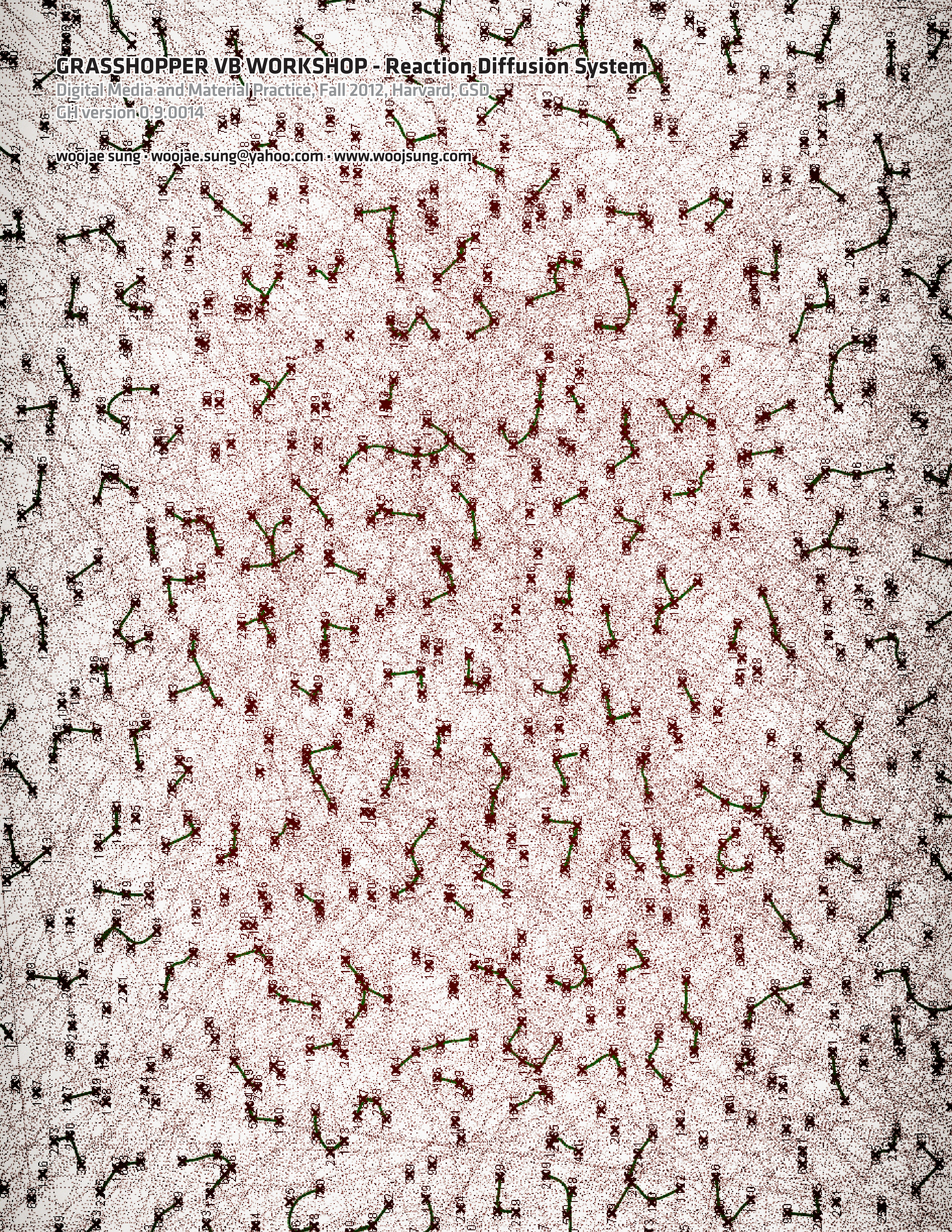
GRASSHOPPER VB WORKSHOP - Reaction Diffusion System

Digital Media and Material Practice, Fall 2012, Harvard, GSD

GH version 0.9.0014

woojae sung · woojae.sung@yahoo.com · www.woojsung.com



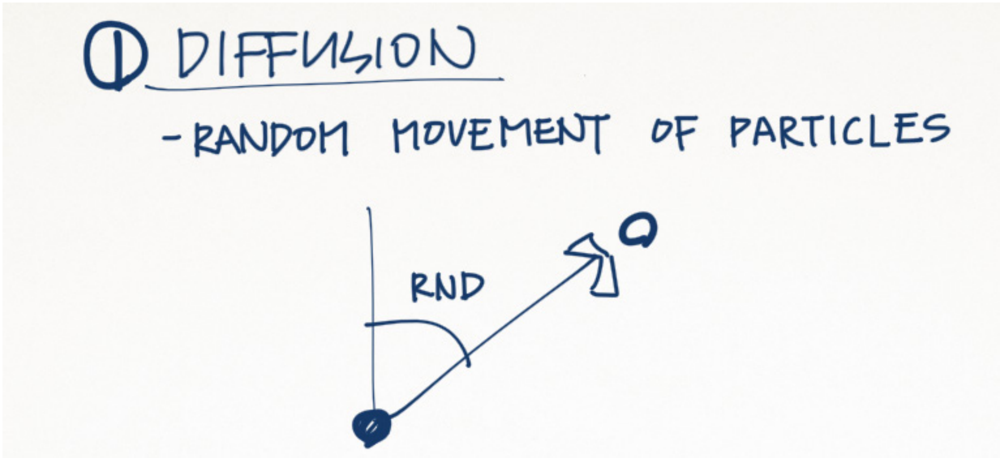
REACTION DIFFUSION SYSTEM

Reaction–diffusion systems are mathematical models which explain how the concentration of one or more substances distributed in space changes under the influence of two processes: local chemical reactions in which the substances are transformed into each other, and diffusion which causes the substances to spread out over a surface in space (from Wikipedia).

Reaction Diffusion system in this material is not describing exactly what it really is. Rather, this is my own interpretation of the system. The system can be simply break down into two main processes; ***Diffusion and Reaction***

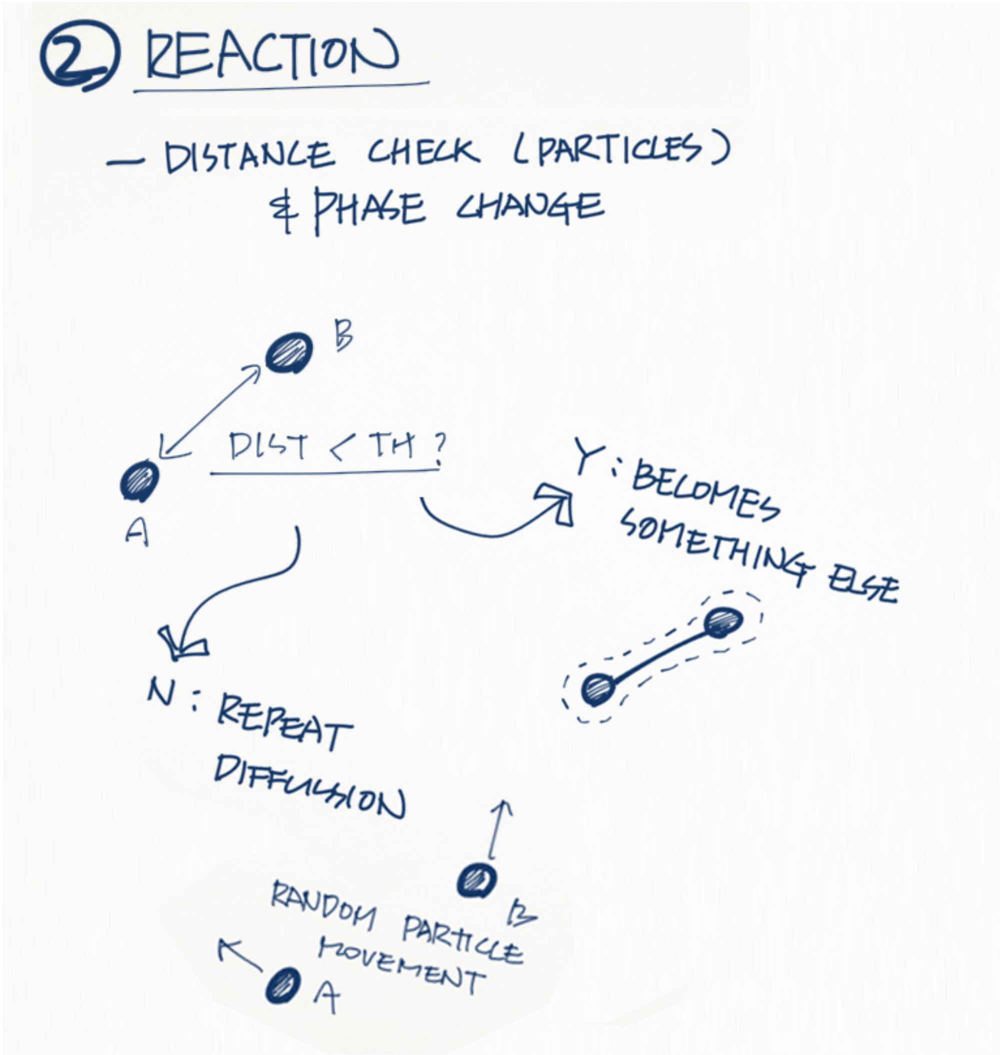
Diffusion (Wandering Particles)

Unlike DLA system's, particles wandering in Reaction Diffusion system don't have any dominant direction. They move in a random fashion spreading over time.

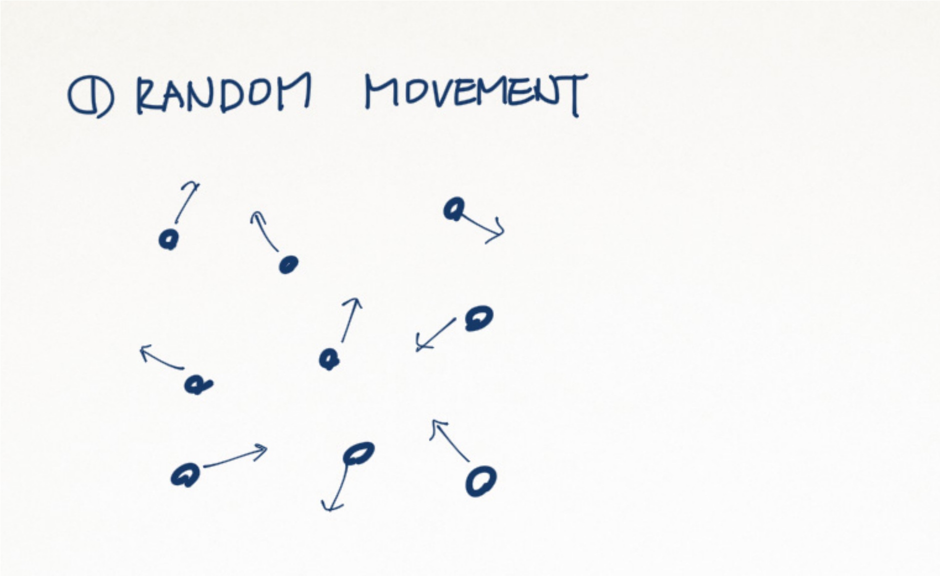


Reaction (Aggregation - Sticking Particles within threshold)

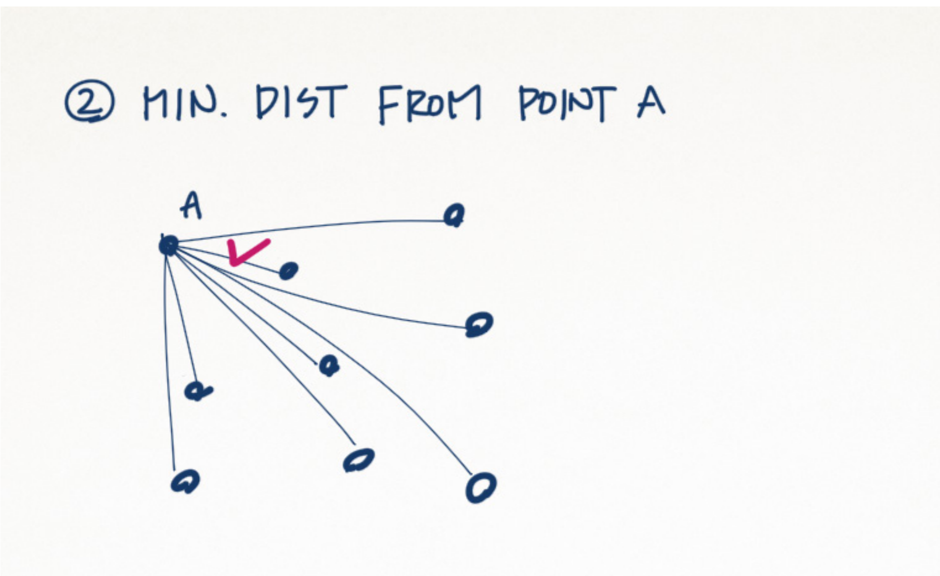
One of the most challenging thing about the system is on its complexity. Unlike DLA system where there is only one moving particle at a time, all particles in the Reaction Diffusion system move randomly all together at the same time. So in each iteration, we have to calculate and check every possible combination of distances between points.
Constantly check if there is a set of two points close enough to each other. If yes, stop wandering and merge the two points into a single line, and stop moving those points. Otherwise, let the particle wander until they get close enough to the threshold.



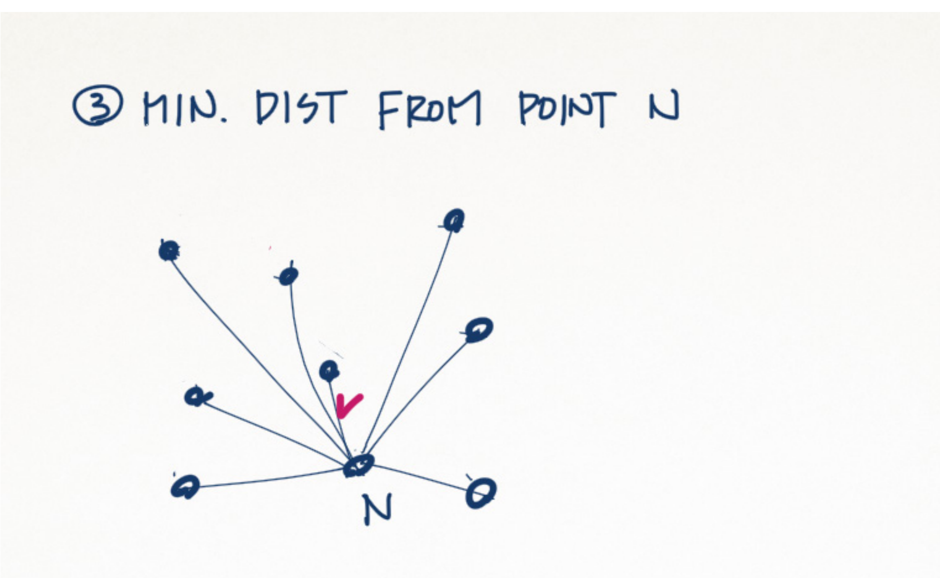
PROCESS



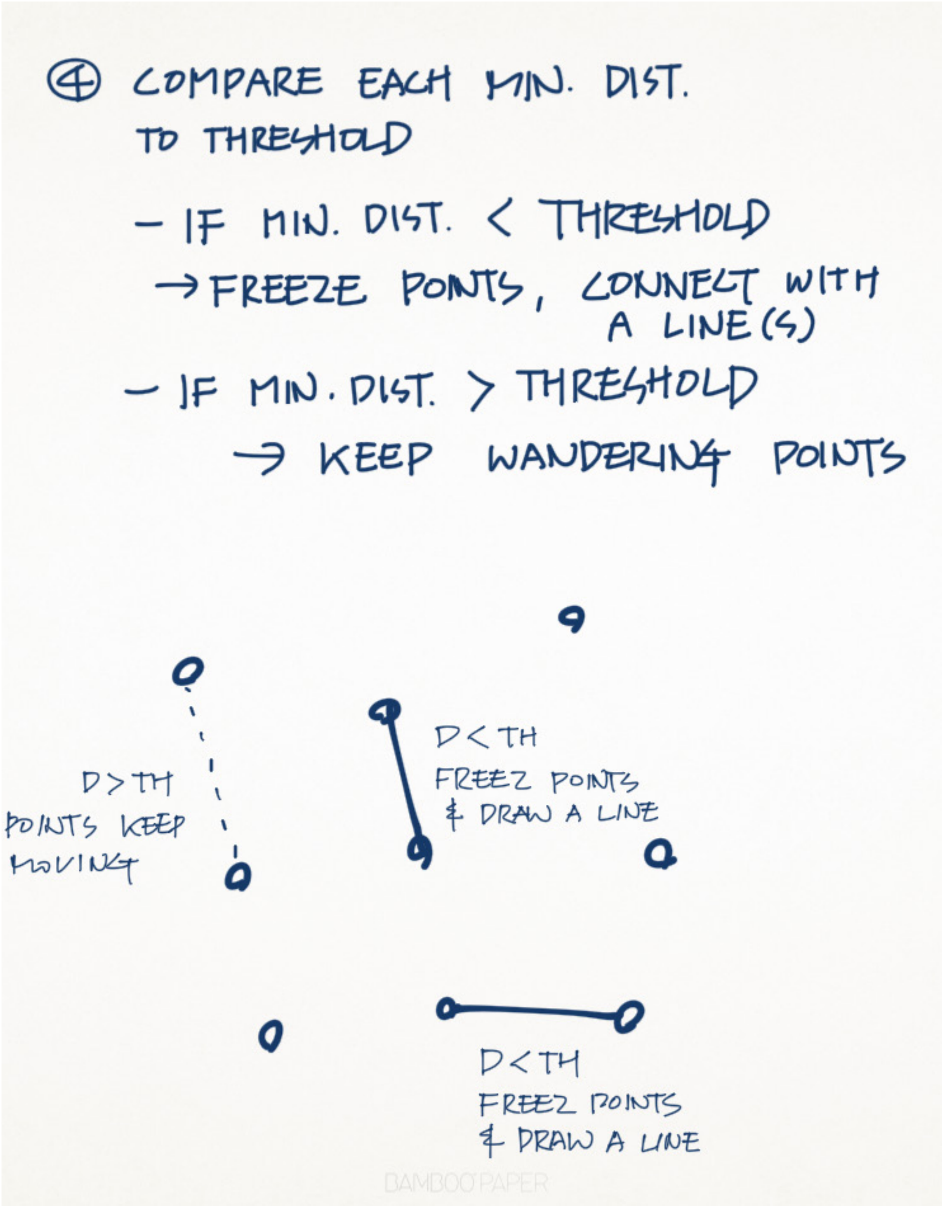
step 01
Wander points / random direction.



step 02
Check all possible cases of distance from the a point (A).

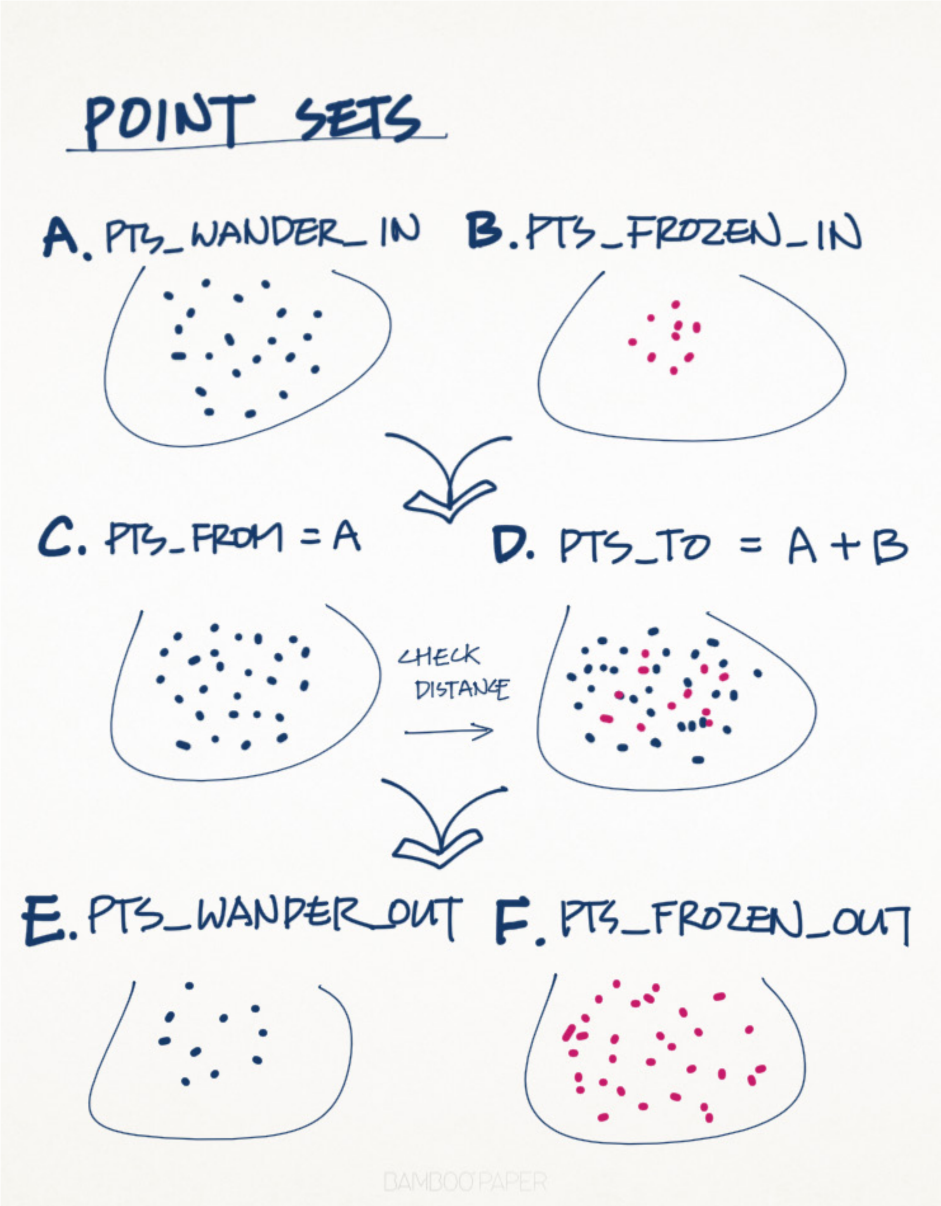


step 03
Check all possible cases of distance from the a point (N).



step 04

Get the list of shortest distances between points and compare the list to threshold to merge points if the number is smaller than the threshold.



step 06

Basically there are 6 sets of points. At a certain point of process, there should be two input point sets, (a)pts_wander_in and (b)pts_frozen_in. Then to check every possible cases of distance between points, we create two additional sets of points, (c)pts_from and (d)pts_to. (c)pts_from is from (a)pts_wander_in and (d)pts_to is a mixture of both (a)pts_wander_in and (b) pts_frozen_in. After checking distances with threshold, we lose some points close enough to other points from (a) or (c), forming (e)pts_wander_out. At the same time, we get some points from (a) or (c) growing size of fronzen point set (f)pts_frozen_out.

Code Reference
http://www.rhino3d.com/5/rhinocommon/