

05-J-04 Delaunay Triangulation

Randomized Incremental Construction

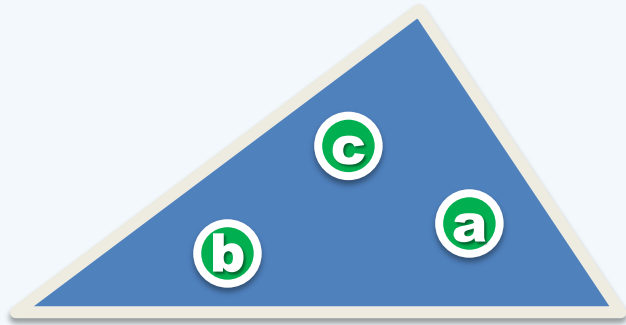
- Point Location

Junhui DENG

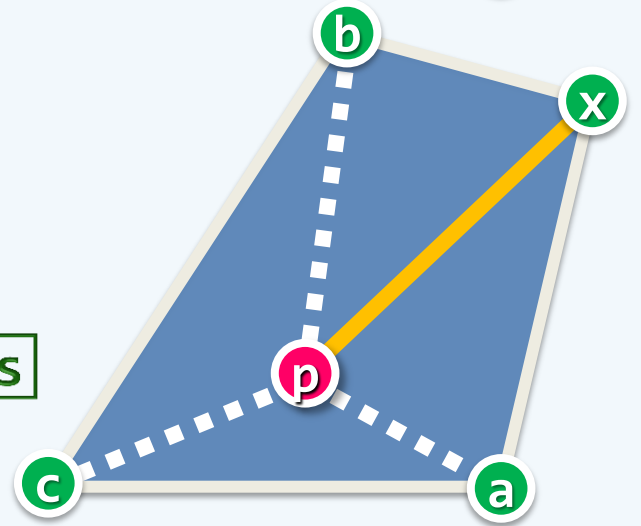
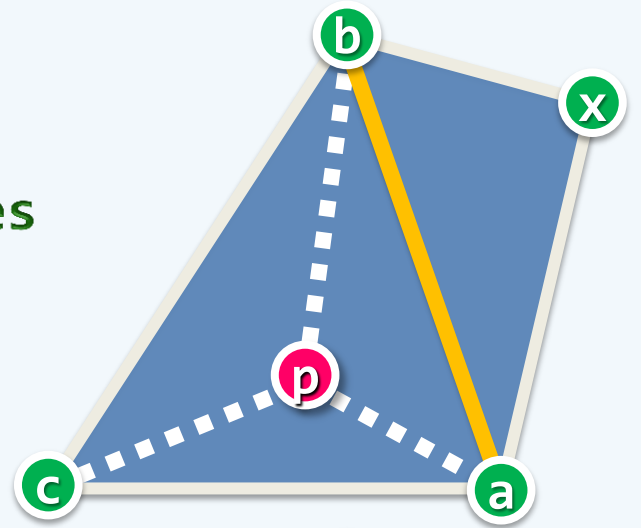
deng@tsinghua.edu.cn

Point Location

- ❖ Remember that we need some way of knowing that
in which triangle each newly inserted point lies



- ❖ Instead of creating advanced data structures,
here we use a simpler method that
puts each of the un-inserted points into buckets
according to the triangle that
it is contained in the current triangulation

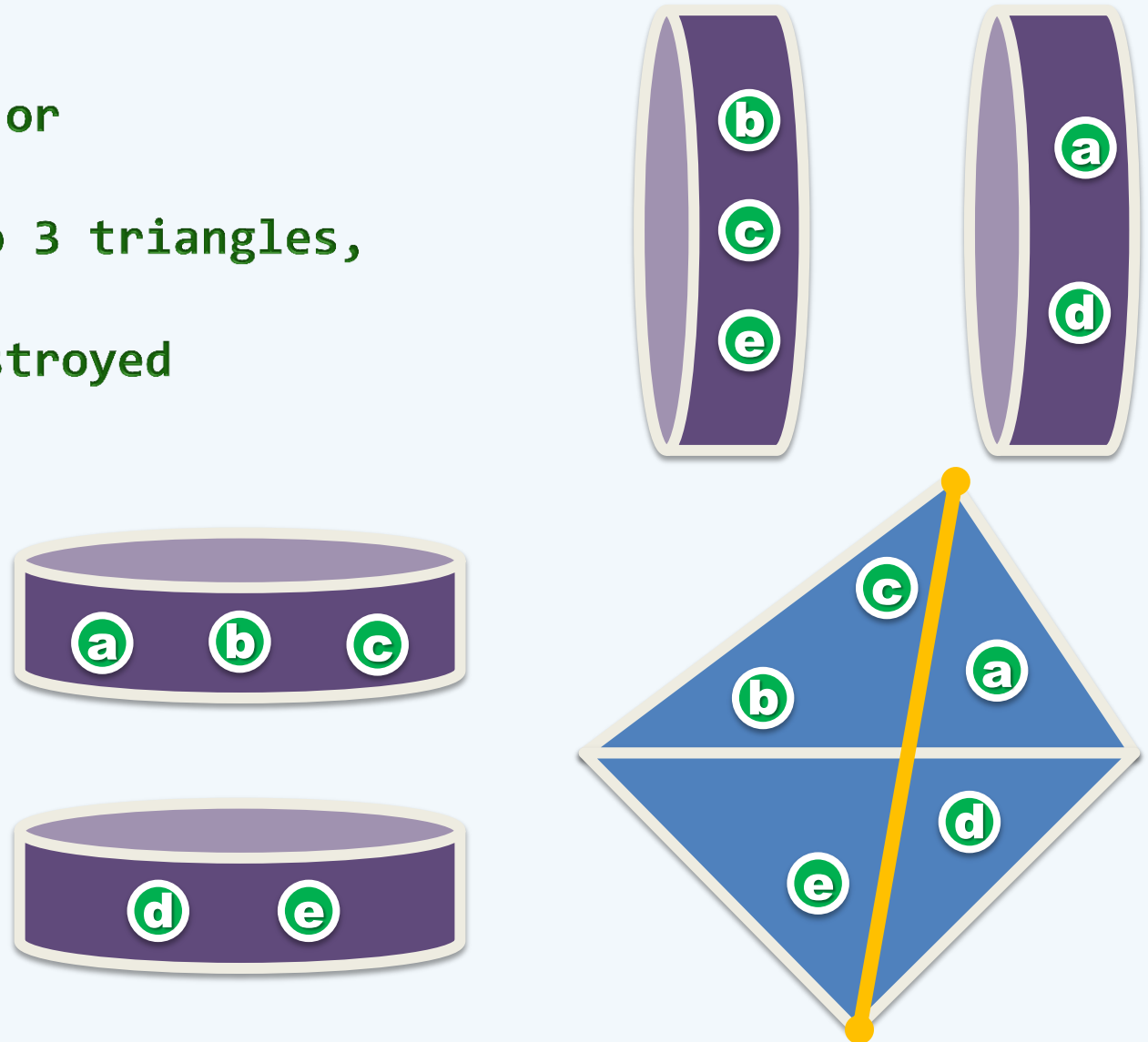


Rebucketing

- ❖ Whenever an edge is flipped, or
when a triangle is split into 3 triangles,
some old triangles are destroyed
and replaced by

a constant number of
new triangles

- ❖ What should we do
when this happens?



Rebucketing

❖ , we

- **lump together** all the sites in the buckets corresponding to the deleted triangles,
- **create** new buckets for the newly created triangles, and
- **reassign** each site into its new bucket

❖ As will be shown later, there are **expected- $\mathcal{O}(1)$** triangles born during each iteration

❖ It costs **$\mathcal{O}(1)$** time to re-bucket **each** site

