Triangulace

Matematická formulace problému

- Dáno: Množina bodu $P = \{p_1, p_2, ..; p_n\}$ v R^2.
- Hledáme: Triangulaci *T nad množinou P.*
- **Definice:**
  - *Triangulace T nad množinou bodu P představuje takové rozdělení roviny, které vytvoří soubor m trojúhelníku* $T = \{t_1; t_2; ..; t_m\}$ *tak, aby platilo:*
    - *Libovolné dva trojúhelníky* t_i , t_j *z T ; (i <> j), mají společnou nejvýše hranu.*
    - *Sjednocení všech trojúhelníku t z T tvoří konvexní obal množiny bodů P.*
    - *Uvnitř žádného trojúhelníku neleží žádný další bod z P.*

Odhady počtu trojúhelníků

- Vztah mezi počtem bodů *n, počtem hran h a počtem ploch t v rovinném grafu (Eulerova věta):*
  - ***n + p = h + 2***
- Pro dokonalou triangulaci T (všechny plochy včetně vnější jsou trojúhelníky) platí:
  - 3p = 2h, h=3/2p
- A tedy
  - n + p = 3/2p +2
  - p = 2n - 4,
  - h = 3n - 6,
- *Pokud vnější plocha nebude trojúhelník, platí nerovnost (ovšem „ne moc vydálená od rovnosti“)*
  - **p <= 2n - 4,**
  - **h <= 3n - 6**

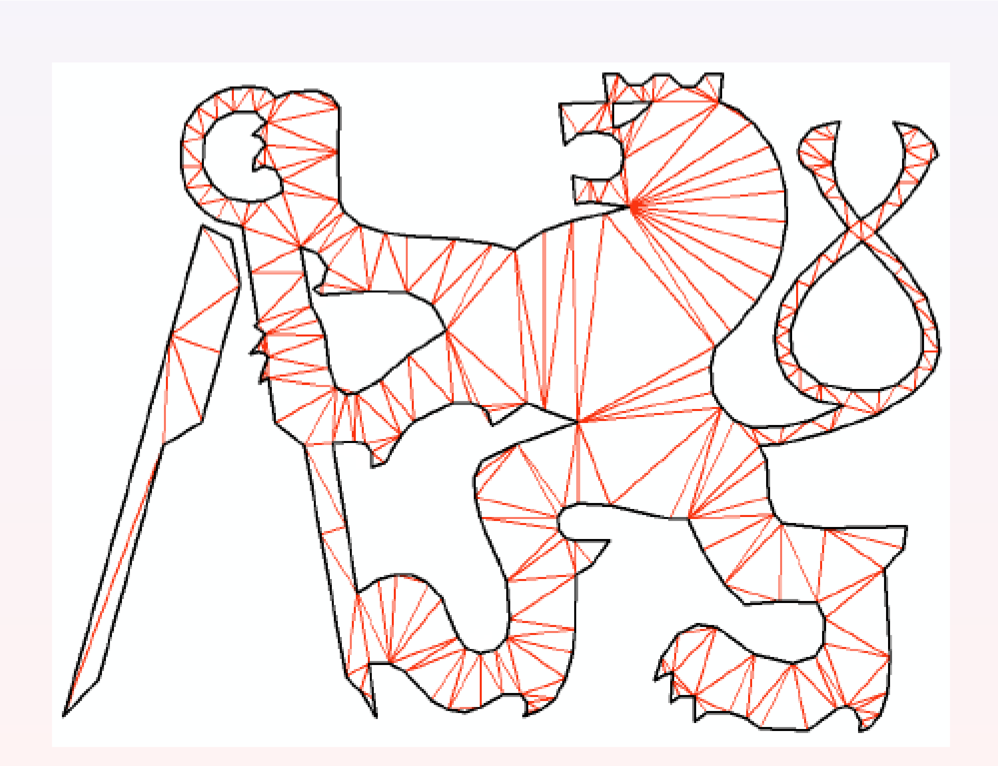
Požadavky na triangulaci T

- Jednoduchost algoritmu, snadná implementace.
- Dostatečná rychlost pro velká *P (n > 1.000.000) bodů, alespoň požadavek na O(n log(n)) algoritmus.*
- Malá citlivost na singulární případy.
- Dobrý tvar trojúhelníkové sítě.
- Některé body v protikladu:
  - jednoduchost implementace x rychlost.

Požadavky na triangulaci

- **Tvar trojúhelníku:**
  - Triangulace by mela produkovat pravidelné trojúhelníky vhodných tvaru (blížící se rovnostranným). Kritérium je důežité při tvorbě DMT, trojúhelníková síť se musí co nejvíce přimykat k terénu.
- **Povinné hrany:**
  - Schopnost vkládat povinné hrany a modifikovat tvar triangulace. Ovlivnění tvaru terénu, vkládání kosterních čar a singularit.
- **Triangulace nekonvexní oblasti:**
  - Schopnost triangulace nekonvexní oblasti či oblasti obsahující díry. V mapách nejsou triangularizovány některé oblasti, např. vodní plochy, budovy.

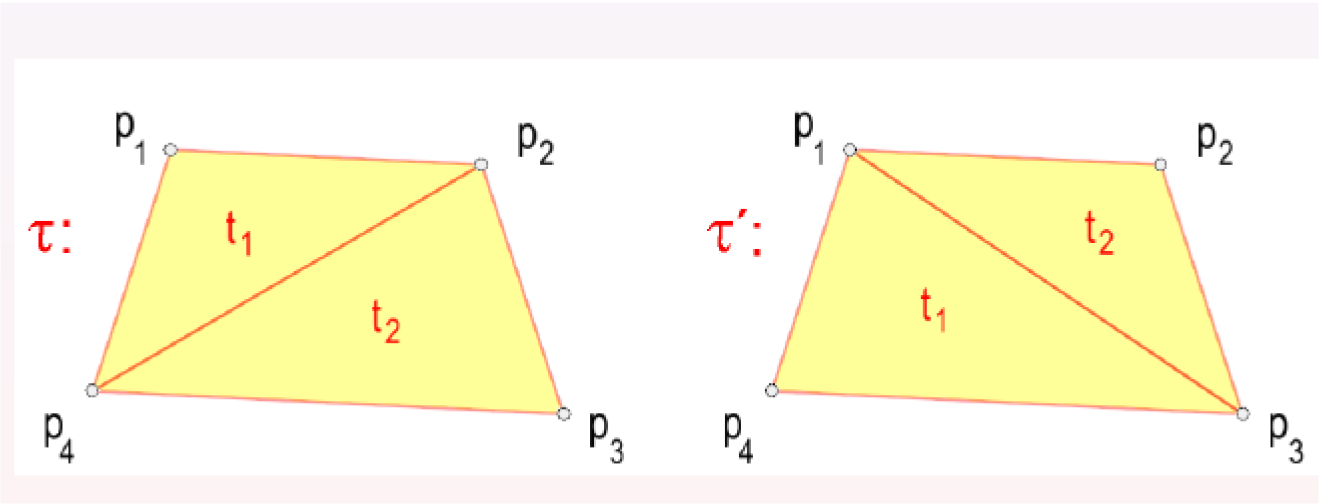
Triangulace nekonvexní oblasti s dírami



Lokální a globální kritéria optimality

- **Lokálně optimální triangulace T :**
  - Každý čtyrúhelník tvořený dvojicí trojúhelníku se společnou stranou je triangularizován optimálně vzhledem k zadanému kritériu. Pro množinu *P existuje více lokálně optimálních triangulací, každá z nich* optimalizuje jiné kritérium.
- **Globálně optimální triangulace T**
  - Všechny trojúhelníky triangulace T optimální vzhledem k zadanému kritériu. Neexistuje jiná triangulace T', která by dosáhla alespoň u jednoho trojúhelníku lepší hodnoty posuzovaného kritéria. Globálně optimální triangulace je současně lokálně optimální.
- **Multikriteriálně optimalizované (kompromisní) triangulace T :**
  - Kombinace několika lokálních či globálních kritérií. Vycházejí obvykle z Delaunay triangulace, která je upravována vzhledem k dalším kritériím. Dlouhé výpočetní casy, doposud nejsou známy efektivní algoritmy, použití genetických algoritmu.

Lokální kritéria



Lokální kritéria

- Mají geometrický podtext, snaha o generování trojúhelníku “rozumných” tvaru.
- Přehled nejčastěji používaných lokálních kritérií:
  - Minimální/maximální úhel v trojúhelníku .
  - Minimální/maximální výška v trojúhelníku *v.*
  - Minimální/maximální poloměr vepsané kružnice *r.*
  - Minimální/maximální poloměr opsané kružnice *R.*
  - Minimální/maximální plocha trojúhelníku *S.*
- Nejčastěji používáno první kritérium (Delaunay triangulace minimalizuje maximální úhel).

Minimaxová kritéria založená na vnitřních úhlech

- **Min-max kritérium:**
- Eliminace trojúhelníku s příliš tupými úhly. Triangulace T (*P) je vzhledem k tomuto*

kritériu na rozdíl od T'(*P) optimální, je–li nejvetší úhel generovaný triangulací T (P)* menší než nejvetší úhel generovaný triangulací T '(*P).*

  - $\alpha_{max} = max(\alpha_i (T))$
  - $\alpha_{max} = max(\alpha'_i (T'))$
  - $\alpha_{max} < \alpha'_{max}$
- **Max-min kritérium:**
- Eliminace trojúhelníku s příliš ostrými úhly. Triangulace T (*P) je vzhledem k tomuto* kritériu na rozdíl od T'(*P) optimální, je–li nejmenší úhel generovaný triangulací T* vetší než nejmenší úhel generovaný triangulací T '(*P).*
  - $\alpha_{min} = min(\alpha_i (T))$
  - $\alpha_{min} = min(\alpha'_i (T'))$
  - $\alpha_{min} > \alpha'_{min}$

Globální kritéria

- Optimalizují geometrické parametry všech trojúhelníku v triangulaci T (*P).*
- Nejčasteji používaná kritéria:
- **Suma délek stran:**
  - Zohledňuje celkovou délku stran *vytvořené triangulace (minimalizace) Σd(hi)→ min*
  - *MWT (Minimum Weight Triangulation).*
  - Pro obecný prípad polohy bodu v R2 nevyrešeno, priblížné rešení genet. algoritmy.
- **Povinné hrany:**
  - Předem definované hrany uvnitř triangulace, tzv. Constrained Triangulation. Taková T (*P) není lokálně optimální.*

Hladová (Greedy) triangulace

- **Vlastnosti triangulace:**
  - Snaží se vytvářet trojúhelníky s nejkratšími stranami, trojúhelníky nemusí splňovat žádnou speciální geometrickou podmínku.
  - Pokud se v *P nevyskytují hrany se stejnou délkou, je triangulace* jednoznacná.
  - Jednoduchá implementace.
  - Velká výpočetní složitost je *O(n3), lze optimalizovat na O(n2 log(n)).*
- **Dusledek:**
  - Síť trojúhelníků často není z tvarového hlediska pěkná do triangulace, k mohou být přidány tvarově nevhodné trojúhelníky.
  - Výpočtní složitost je příliš veliká
  - V kartografii není příliš často používána.
  - Výsledná triangulace se blíží MWT.

Algoritmus hladové triangulace

Algoritmus 1: Greedy Triangulation (S, Q)

Opakuj pro $\forall p_i$, $i \in \langle 1, n \rangle$:

    Opakuj pro $j \in \langle i+1, n \rangle$:

        Vytvoř hranu $e = (p_i, p_j)$.

        Pro e urči $d(p_i, p_j)$ a ulož do Q.

Setřiď Q dle d.

Odeber $Q[0]$ a přidej do $\mathcal{T}$.

Dokud Q není prázdná:

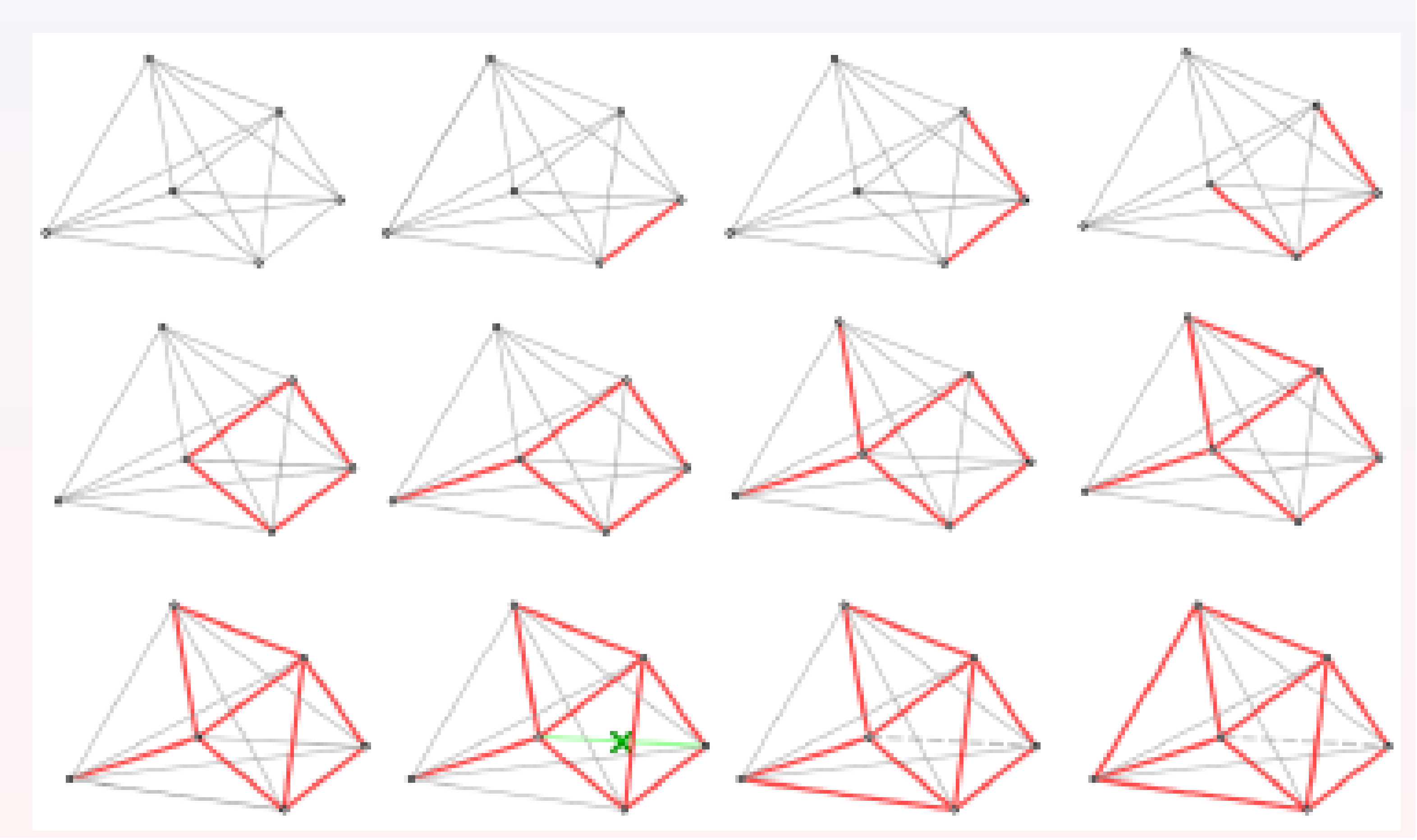
    $e =$pop(Q).

    Opakuj pro $\forall e_i \in \mathcal{T}$:

        test, zda e protíná $e_i \in \mathcal{T}$.

    Pokud e neprotíná žádné $e_i \in \mathcal{T}$:

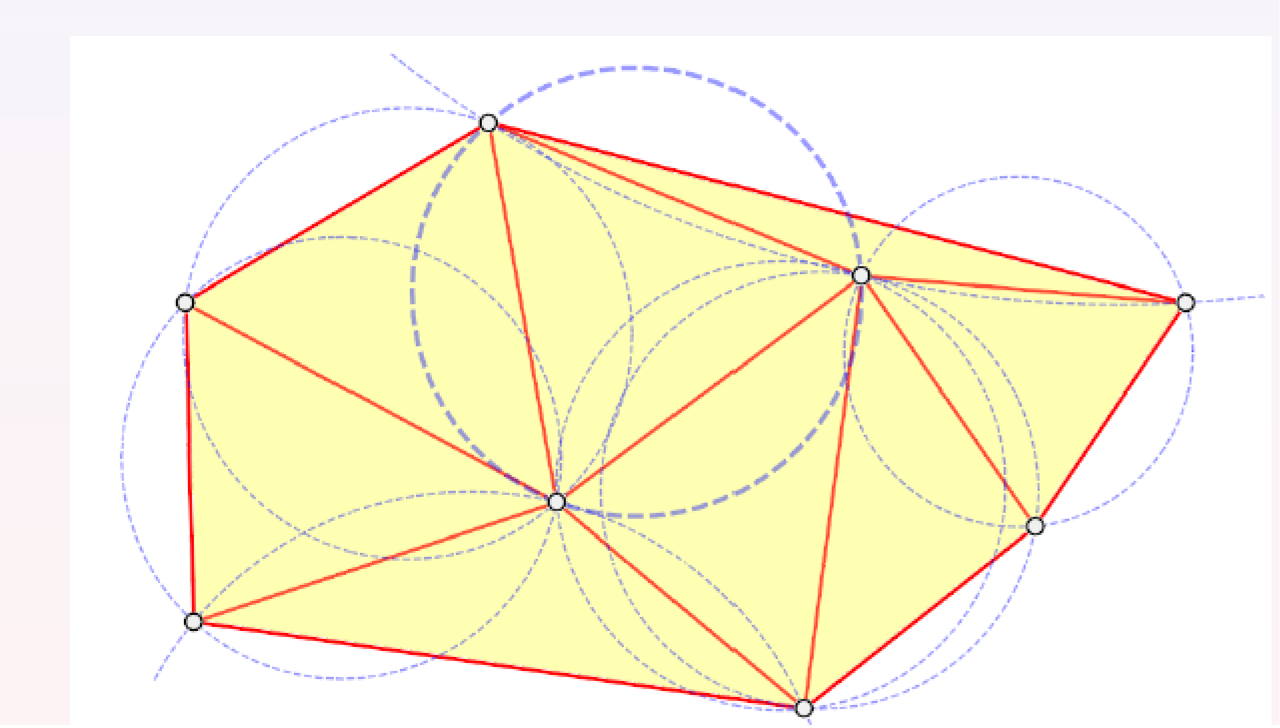
Algoritmus hladové triangulace



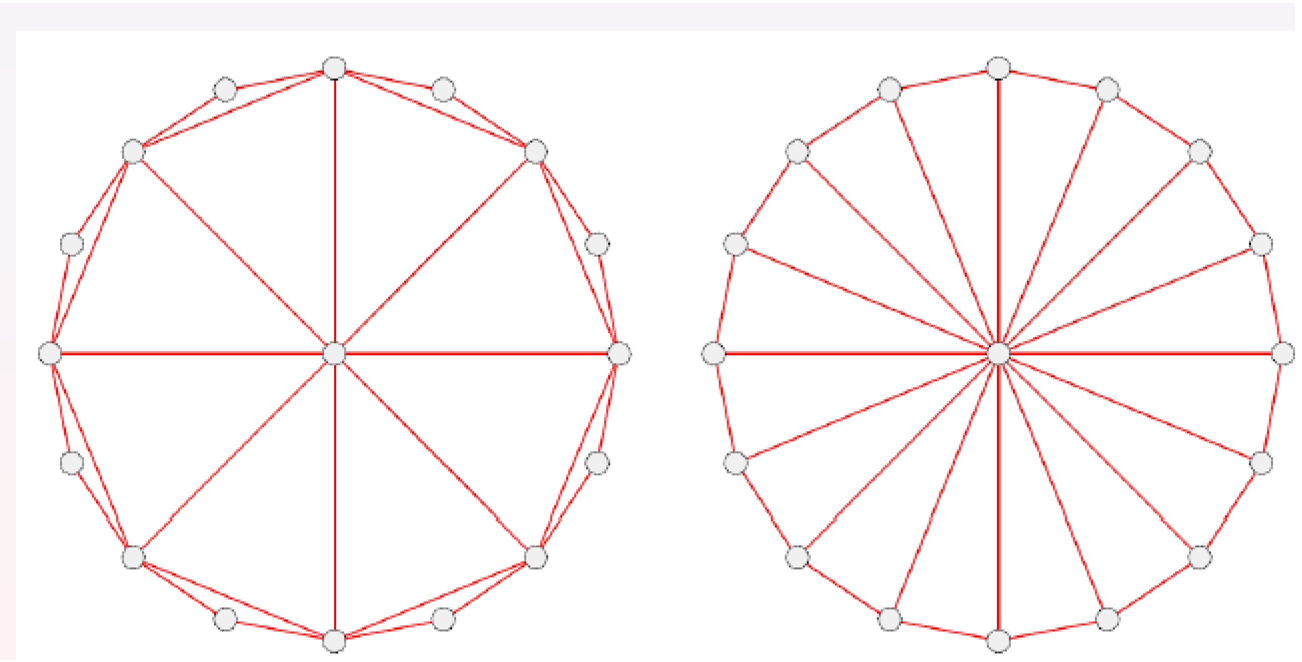
Deleunay triangulace

- Nejčastěi používaná triangulace, v oblasti GIS de-facto standart.
- *Uvnitř kružnice opsané libovolnému trojúhelníku neleží žádný jiný bod množiny P.*
- DT *minimalizuje maximální úhel v každém trojúhelníku.*
- DT *je lokálně optimální i globálně optimální vůči kritériu minimálního úhlu.*
- DT *je jednoznačná, pokud žádné čtyři body neleží na kružnici.*
- Výsledné trojúhelníky se při porovnání ze všemi známými triangulacemi „nejvíce blíží“ rovnostranným trojúhelníkům.

Deleanuy triangulace



Srovnání GT a DT



Metody konstrukce DT

- Metody přímé konstrukce DT :
  - Lokální prohazování.
  - Inkrementální konstrukce.
  - „Rozděl a panuj“.
- Nepřímá konstrukce: přes Voroného diagram

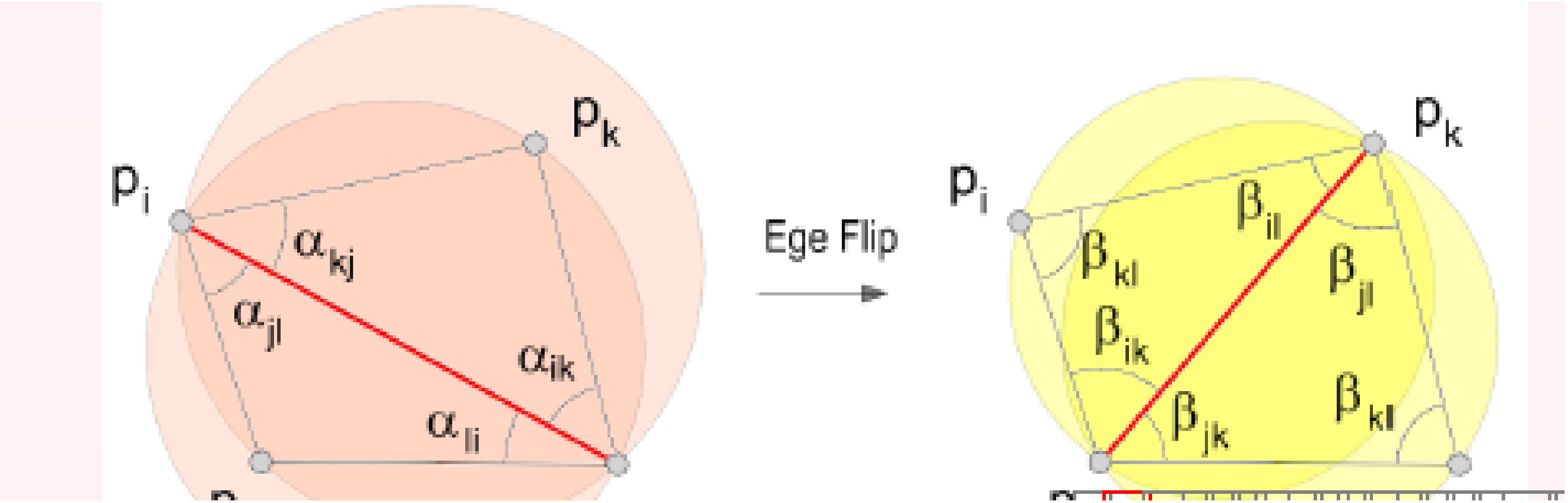
Metoda lokálního prohazování

- Převod libovolné triangulace T na DT .
- Prohazování nelegálních hran v dvojicích trojúhelníku tvořících konvexní ctyrúhelník.
- Složitost algoritmu je *O(N^2), nutno připocítat složitost yákladního triangulačního* algoritmu.
- Lze použít vzhledem k libovolnému lokálnímu kritériu.

Legalizace

- Nechť *P je množina bodů pi ; pj ; pk ; pl tvořící vrcholy konvexního čtyrúhelníka.*
- Edge Flip = prohození diagonály čtyrúhelníku (swap), tj. prechod z DT (*P) na DT ‘(P)*
- Výsledkem je stav, kdy jsou oba trojúhelníky **legální, tj. lokálne optimální vzhledem ke** kritériu vnitřního úhlu (minimalizace maximálního úhlu).
- Tato operace je opakovaně prováděna nad všemi trojúhelníky.
- *Je* nazývána **legalizací.**

Legalizace



p_i
p_k
α_{kj}
α_{jl}
α_{ik}
α_{li}
Ege Flip
p_i
p_k
β_{il}
β_{kl}
β_{jl}
β_{ik}
β_{kl}
β_{jk}

Algoritmus lokálního prohazování

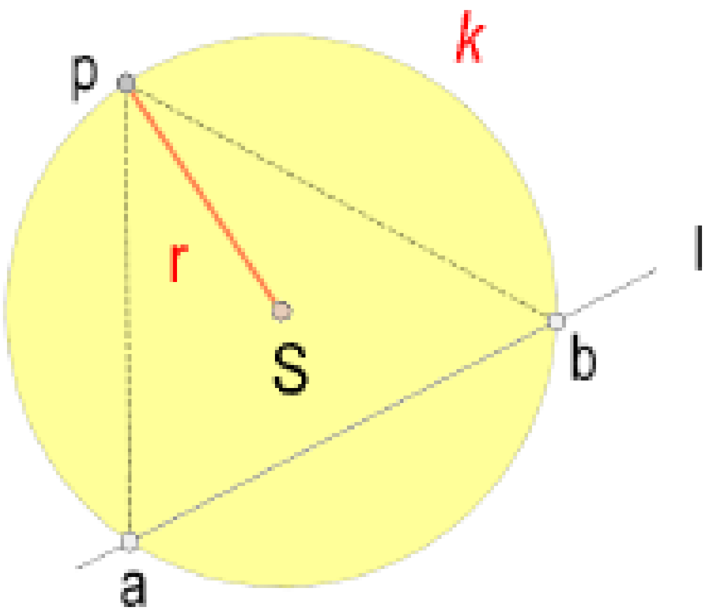
Algoritmus 2: Delaunay Triangulation Local(P)

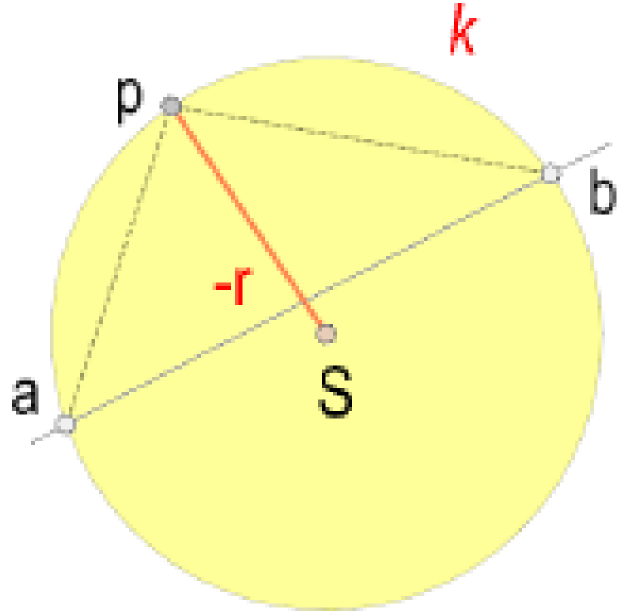
Vytvoř pomocnou triangulaci $T(P)$.
legal=false
while $T(P)$!legal
    legal=true;
    Opakuj pro $\forall e_i \in T(P)$
        Vezmi hranu $e_i \in T(P)$
        Nalezni trojúhelníky t_1, t_2 incidující s e_i.
        if $(t_1 \cup t_2)$ konvexní a nelegální
            Legalize (t_1, t_2).
            legal=false;

Algoritmus přímé (inkrementální) konstrukce DT

- Založen na postupném přidávání bodu do již vytvořené DT .
- Nad existující Delaunayovskou hranou *e = p1; p2 hledám takový bod p, který* má od *p1; p2 minimální Delaunay vzdálenost dD(p1p2; p).*
- Každá Delaunayovská hrana je orientována, bod *p hledáme pouze vlevo od ní.*
- Do DT přidány hrany trojúhelníku (*p1; p2; p).*
- Není-li bod *p nalezen je hrana p1; p2 na hranici konvexního obalu množiny bodů P.*
- Složitost je $O(n^2)$*, použitím vhodných datových struktur lze vylepšit na O(n.ln n), při jistých podmínkách na pravidelné rozložení vstupní množiny bodů dokonce na O(n).*

Deleanuy vzdálenost



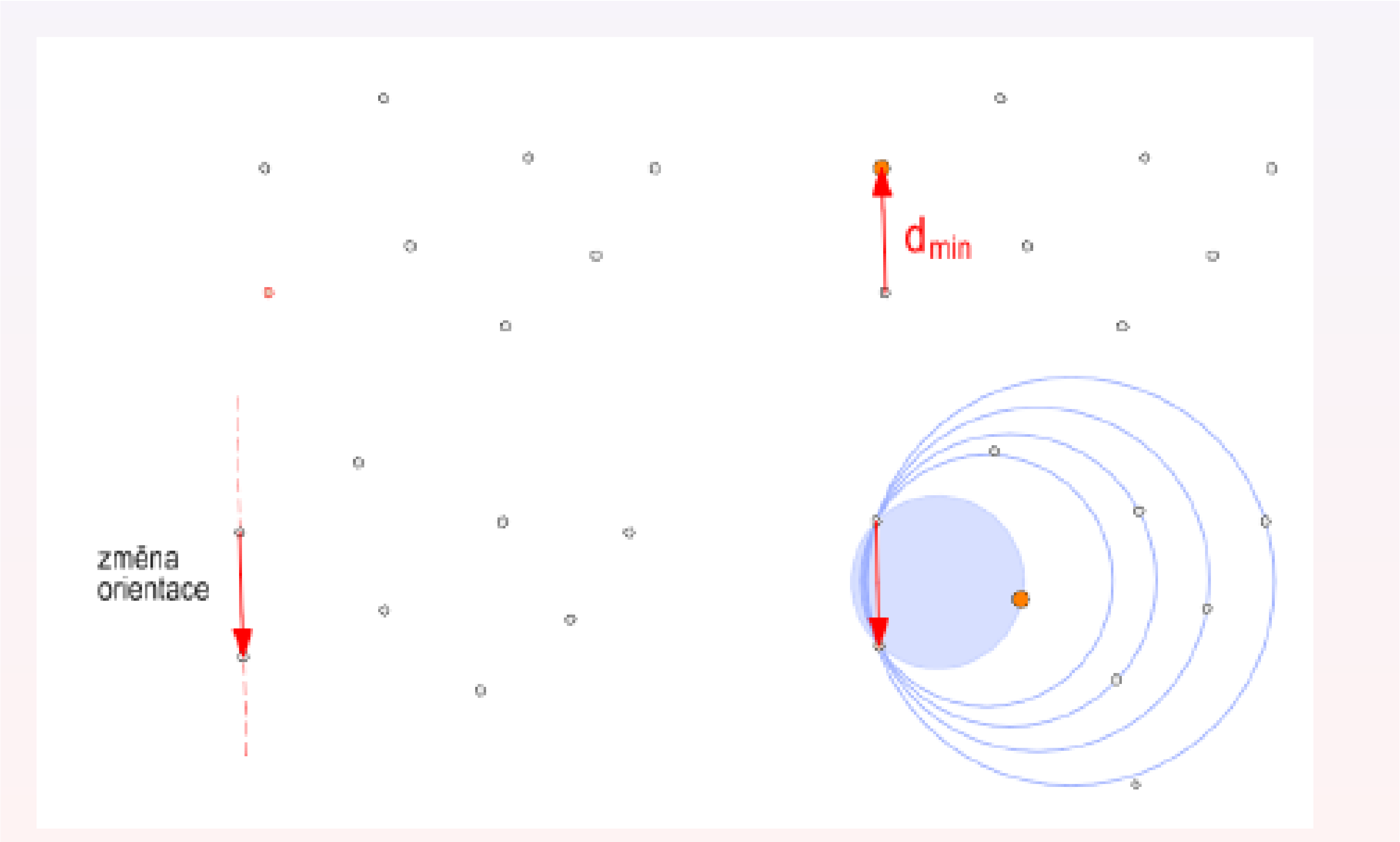


Algoritmus inkrementální konstrukce

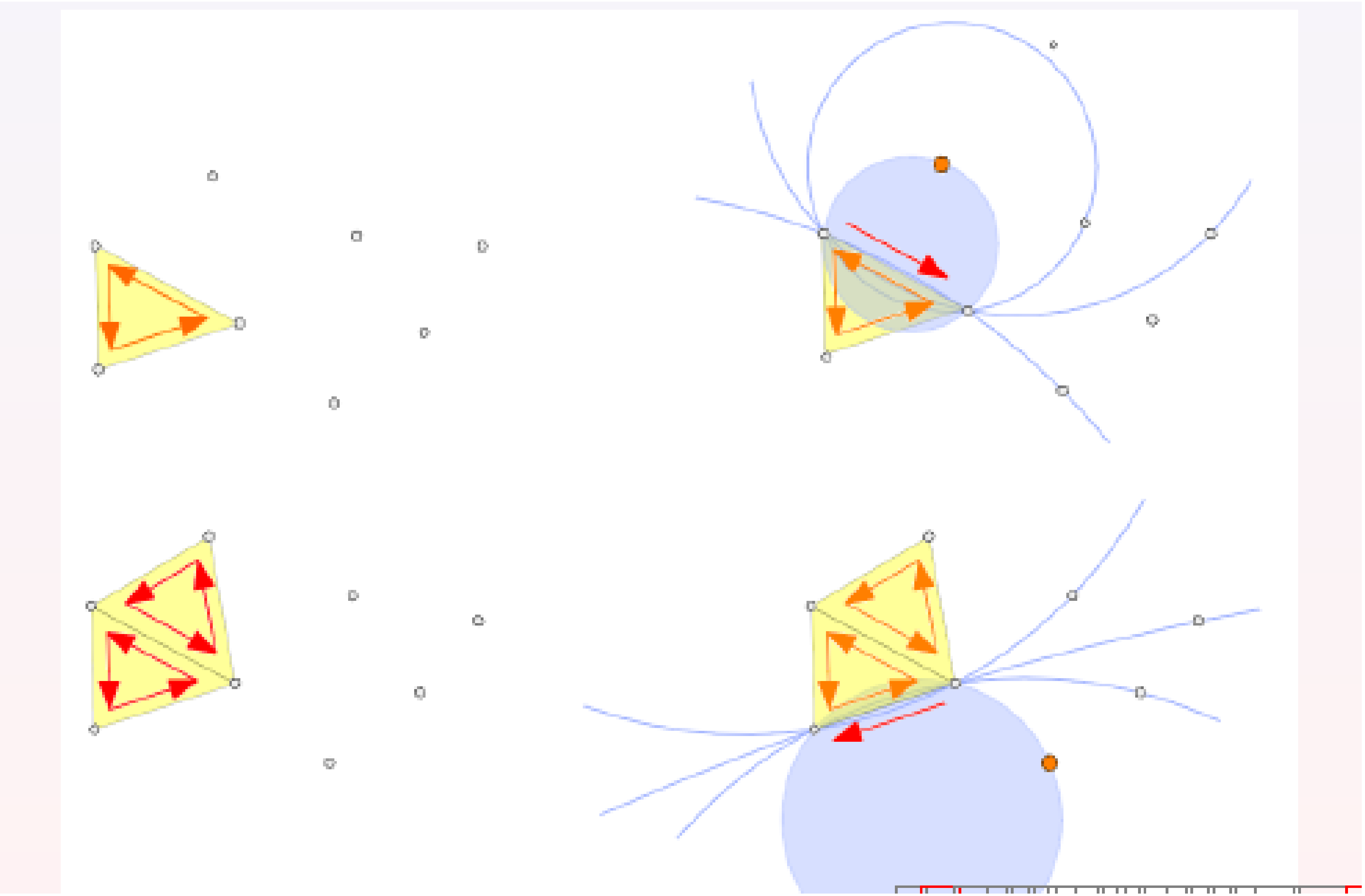
Algoritmus 2: Delaunay Triangulation Incremental (S, AEL, $\mathcal{DT}$)

$p_1 =$ náhodný bod z P, $p_2 =$nejbližší bod k p_1.
Vytvoř hranu $e = \overline{p_1 p_2}$
$p = d_D(e)$, bod s nejmenší Delaunay vzdáleností vlevo od e.
Pokud $p = NULL$, prohoď orientaci $e = \overline{p_1 p_2} \Rightarrow e = \overline{p_2 p_1}$.
$e_2 = \overline{p_2 p}$, $e_3 = \overline{p p_1}$.
Add e, e_2, e_3 do AEL.
while AEL not empty do:
    $e = \overline{p_1 p_2}$ první hrana AEL.
    Změna orientace hrany $e = \overline{p_1 p_2} \Rightarrow e = \overline{p_2 p_1}$.
    Bod p s nejmenší Delaunay vzdáleností $d_D(e)$ vlevo od e.
    if $p! = NULL$:
        $e_2 = \overline{p_2 p}$, $e_3 = \overline{p p_1}$.
        Add e, e_2, e_3 do AEL.
        Add e do $\mathcal{DT}$.
    pop (e)

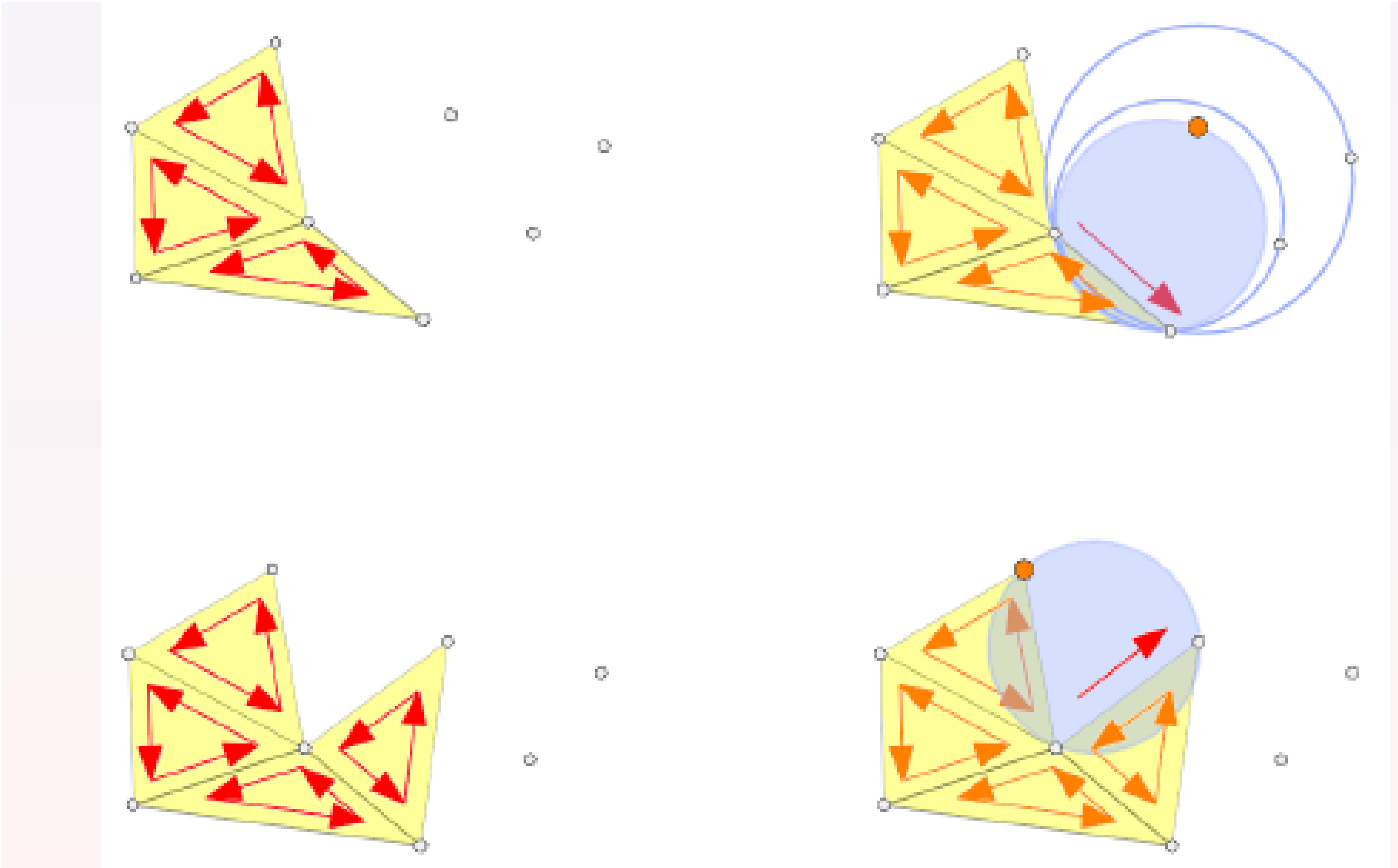
Inkrementální konstrukce



Inkrementální konstrukce



Inkrementální konstrukce



Rozděl a panuj (používá ATLAS)

- Vstupní množina bodů se rozdělí na menší části, každá bude triangulována samostatně
- Výsledné triangulace se na svárech spojí a legalizují
- Při použití výpočetně složitějších algoritmů se výpočet může radikálně zrychlit
- Může vést k paradoxům --- zvětšení počtu bodů způsobí větší dělení, horší triangulaci a méně přesný DMT