

Kinyó Kincső

## SPANNEREK

Szakdolgozat

Matematika BSc, alkalmazott matematikus szakirány

Témavezető:

Pálvölgyi Dömötör



EÖTVÖS LORÁND TUDOMÁNYEGYETEM

TERMÉSZETTUDOMÁNYI KAR

---

Budapest, 2024

# Tartalomjegyzék

<b>1. Bevezetés</b>	<b>3</b>
1.1. Definíciók . . . . .	4
<b>2. Algoritmusok</b>	<b>6</b>
2.1. Mohó algoritmus . . . . .	6
2.2. Yao-, $\Theta$ -gráf . . . . .	10
2.3. WSPD . . . . .	13
2.4. Delaunay-gráfok . . . . .	16
<b>3. Összefoglalás, kitekintés</b>	<b>27</b>
3.1. Az algoritmusok összehasonlítása . . . . .	27
3.2. Tárgyak mozgatása . . . . .	29

# 1. fejezet

## Bevezetés

A körülöttünk lévő világ tele van különböző hálózatokkal, rendszerekkel, melyeket a természetben fedezünk fel, vagy akár mi magunk generálunk. Ilyenek például az úthálózatok egy adott országban vagy csak egy kisebb területen, amelyeket előzetes tervezések alapján szeretnénk optimálisan megépíteni. Tehát adott néhány város, melyeket összekötnénk utakkal, hogy mindenholnan el lehessen jutni mindegyik másik városba a lehető legrövidebb úton. Ezt egy gráffal tudjuk optimálisan szemléltetni, ahol a csúcsok a városok, az utak pedig az őket összekötő élek lesznek. Akár mindegyik városból indíthatunk egy-egy utat mindegyik másikba, és akkor a lehető leggyorsabban jutnánk el mindenhova. Tehát az utakat egy teljes gráf élei mentén építenénk meg. Azonban az építkezések költségét is figyelembe kell vennünk, hiszen nem mindegyik út ugyanolyan hosszú, és kerül ugyanannyiba, valamint nem szeretnénk az összes lehetséges utat megépíteni. Tehát a teljes gráf koncepcióját ebben az esetben elvethetjük. A költségek optimalizálására pedig a minimális költségű feszítőfa egy válasz, melynek megtalálására több algoritmus is ismert. Ekkor előfordulhat, hogy két város közt elég nagy a távolság az úthálózatunkban, de a valóságban igen közel vannak egymáshoz. Valahol a kettő között keressük a választ, azaz mindenholnan mindenhova eljussunk „nem túl nagy” távolságon belül, de ne kelljen „túl sok” utat megépítenünk. Ezeket a „nem túl nagy” és „nem túl sok” jelzőket szeretnénk formalizálni matematikai eszközökkel. Például egy aránnyal, a „valós távolság” és az úthálózatbeli távolság között. Valamint a gráfunkat is pontosan definiálnunk kell a következőkben, hogy az itt kifejtett kiindulási problémánkra választ adhassunk. Ezt a típusú gráfot spannernek fogjuk hívni, mellyel könnyen modellezhetjük az előbb felvetett problémát. Dolgozatom célja, hogy bemutassam a spannereket, különböző konstrukciókat, melyekből spannereket kapunk, és az algoritmusokat, melyekkel ezeket megkaphatjuk. Valamelyik algoritmus ki-

fejezetten arra szolgál majd, hogy spannert építsünk, másíknál pedig utólag látjuk be, hogy spannert kaptunk az algoritmusunkkal.

## 1.1. Definíciók

Vegyünk egy  $G = (V, E)$  teljes gráfot, és az élein értelmezett súlyfüggvényt:

$$w : E \rightarrow \mathbb{R}^+.$$

Itt nyilván az élet jelölhetjük a két végpontjával is:  $w(uv)$ , ami az  $u$  és  $v$  közt futó él súlyát jelzi. Egy  $P$  úton a súlyok összege legyen:

$$W(P) := \sum_{e \in P} w(e).$$

Tekintsük meg még legrövidebb utakat az adott  $G$  gráfban, mert ez fontos lesz a spannerek definiálásához:

$$d_G(u, v) := \min_{P \text{ } u,v \text{ út } G\text{-ben}} W(P)$$

**1.1.1. Definíció.** Egy  $G$  teljes gráf  $G' = (V, E')$  részgráfját  *$t$ -spannernek* nevezzük, ha minden  $u, v$  csúcspárra létezik  $G$ -ben egy út közöttük, amelyre:

$$d_{G'}(u, v) \leq t d_G(u, v).$$

Tehát a spanner a teljes gráf olyan részgráfja, hogy két pont között legfeljebb  $t$ -szer annyi utat kell megtenni, mint ahogy azt optimális esetben tennénk. Úgy is fogalmazhatunk, hogy egy spanner közelíti a teljes gráfot lehetőleg lineáris mennyiségű éllel. Ezt szeretnénk majd elérni.

A  $t$  érték elnevezése: **dilatáció, nyújtási tényező**. Ezt az értéket minden spannernél egy előre megadott, a gráf méretétől független konstansként kezeljük, mint egyfajta inputot. Egy adott  $t$ -spannerhez a következő fogalmakat társíthatjuk még: a  $G'$  **súlyát** úgy kapjuk meg, hogy összeadjuk az  $E'$ -ben lévő élek súlyait. Ezt is jelöljük  $W(G')$ -vel, hasonlóan az utak összsúlyához. A **spanner mérete** pedig

$$S(G') := |E'|$$

legyen. Egy spannert ritkának mondhatunk az élek, és könnyűnek a súly tekintetében. Például, a spannernek lineárisan sok éle van a csúcsokhoz képest.

Még egy fontos fogalom ehhez a témakörhöz az  $MST(G)$ , azaz „minimum (weight) spanning tree”, amelynek a lehető legkisebb a  $W(G')$  értéke  $G$  spannerjei közül, tehát az élek súlyösszege. Csak annyi éle van, hogy az egész gráf még összefüggő maradjon, tehát egy fa lesz. Azonban az ehhez tartozó  $t$  elég rossz:  $\Omega(n)$ . Ehhez tudjuk majd hasonlítani egy tetszőleges spanner méretét, hiszen mindkét szempontból ez a legkritkább, és célunk lehet, hogy ehhez közeli méretű, súlyú spannereket gyártsunk mi is különböző algoritmusokkal.

A fent bevezetett definíciónak többféle változatát, általánosítását is tekinthetjük. Például nem kell feltétlenül a kiinduló  $G$  gráfnak egy teljes gráfnak lennie. Ekkor a  $t$ -spanner, azaz  $G'$  lényegében egy tetszőleges  $G$  gráf részgráfja, amelyben minden két csúcs közti legrövidebb út  $G'$ -ben sosem haladja meg a  $G$ -beli legrövidebb út  $t$ -szeresét.

Másik konkrét példa lehet, hogy úgynevezett geometriai gráfokat vizsgálunk, amelyekben  $V \subset R^d$  csúcsok vannak, és a csúcsokat egyenes szakaszok kötik össze, amelyek hosszát euklideszi távolsággal mérjük. Ezen geometriai gráfokra is nézhetünk  $t$ -spannereket, lehet külön vizsgálni, hogy két dimenzióban az adott gráf síkbarajzolható-e? A két dimenziós geometriai gráfokkal tudjuk modellezni például az úthálózatokat, és azok optimális megépítését.

Mindezen példák mellett mi az elsőként megemlített definíciót fogjuk használni.

A továbbiakban a szakdolgozatom szerkezete a következő lesz: először négy algoritmust fogunk vizsgálni, melyek spannereket generálnak a 2. fejezetben. Egy mohó algoritmust, mely elég intuitív (2.1). Két algoritmusunk geometriai megoldásokon alapszik, egyik a Delaunay-gráfokat használja (2.4), a másikat pedig  $\Theta$ - vagy Yao gráfnak fogjuk hívni (2.2). Utolsó algoritmusunkban pedig a gráf csúcsait fogjuk szintezni, a szintek és a csúcsok távolságai alapján alkalmas éleket húzunk be (2.3)

## 2. fejezet

# Algoritmusok

### 2.1. Mohó algoritmus

Az első algoritmusunk, mely egy spanner előállítására szolgál, elég egyszerű lépésekből áll, könnyen érthető, hogy mit csinál. Erre többen egymástól függetlenül rájöttek, nem feltétlenül a spannerek témakörében: Bern ([Be]) valamint Althöfer és munkatársai ([?]).

#### Az algoritmus

Az algoritmusunk inputja a  $V$  csúcshalmaz és az elérni kívánt  $t$ , tetszőleges pozitív valós. A spanner kezdő élhalmaza pedig legyen  $E' := \emptyset$ . Az algoritmus során végignézünk minden  $uv$  élet  $E$ -ből, súlyuk szerint nem csökkenő sorrendben. Egy adott élre, ha  $d_{G'}(u, v) > td_G(u, v)$ , akkor az  $uv$  élet adjuk hozzá  $E'$ -hez. Outputként az algoritmus a  $G' = (V, E')$  gráfot fogja visszaadni. Az algoritmus tehát nem köti ki, hogy az eredeti  $G$  gráfnak teljesnek kell lennie. Ha nem teljes gráf van, akkor egy  $E$  halmaz is része az inputnak, és csak ezeket az éleket nézi végig az algoritmus, nem az összes lehetségeset. Most pedig nézzünk néhány állítást a kapott  $G'$ -ről.

**2.1.1. Állítás ([ADDJS]).**  $G'$  egy  $t$ -spanner lesz az előre megadott  $t$ -re.

**Bizonyítás.** Vegyünk egy  $uv$  élet  $G \setminus G'$ -ben, amit az algoritmus elutasított, tehát nem teljesül rá a feltétel, amit ellenőriz az összes élre. Azaz létezik egy  $P$  legrövidebb út az aktuális  $G'$ -ben, melyre  $W(P) \leq td_G(u, v)$ .

Legyen két tetszőleges csúcs  $V$ -ből  $a$  és  $b$ . Erre a két csúcsra vegyük azt a  $Q$  utat  $G$ -ben, amely a legkisebb súlyú, tehát a legrövidebb  $ab$  út, és cseréljünk le minden olyan élet, ami nincs  $G'$ -ben a hozzá tartozó  $P$  útra, amely már  $G'$ -ben halad. Ekkor a kapott  $Q'$  út hossza

megnőtt, mert  $Q$  volt a legrövidebb  $ab$  út, valamint minden cserénél csak növeltünk, de minden élet egy maximum  $t$ -szer akkora összsúlyú útra cseréltünk. Mint ahogy azt elején láttuk. Tehát  $W(Q) < W(Q') \leq tW(Q)$ , ezzel be is láttuk az állítást.  $\square$

Ez a bizonyítás nem csak akkor működik, ha  $G$  egy teljes gráf, a bizonyításban sem feltételezzük ezt, hogy más esetekre is működjön.

**2.1.2. Állítás ([ADDJS]).** *Legyen  $C$  egy tetszőleges kör  $G'$ -ben. Erre igaz, hogy  $S(C) > t + 1$ , azaz a kör hossza legalább  $t + 2$ .*

**Bizonyítás.** Indirekt tegyük fel, hogy van olyan  $C$ , amire  $S(C) \leq t + 1$ . Legyen  $uv$  az az él  $C$ -ben, amit az algoritmus utolsónak választott be, tehát ennek az egyik legnagyobb a súlya a körben (lehetnek azonos súlyú élek is). Amikor az algoritmus ezt az élet vizsgálja,  $C$  többi éle már benne van az aktuális  $G'$ -ben, azaz legfeljebb  $t$  db él legfeljebb  $w(u, v)$  súllyal. Ekkor tehát  $G'$ -ben van egy út  $u$  és  $v$  között, amelynek hossza  $\leq tw(u, v)$ . Tehát az  $uv$  élet az algoritmus nem fogja bevenni  $G'$ -be, ellentmondásra jutottunk.  $\square$

Tehát a  $G'$ -ben a girth értéke, a legrövidebb kör hossza  $g \geq t + 2$ . A Moore-korlát segítségével a következő becslést kapjuk az élek számára:

$$S(G) \leq n^{1 + \frac{1}{\lfloor \frac{g-1}{2} \rfloor}} \leq n^{1 + \frac{2}{t-1}}$$

Hiszen:

$$\lfloor \frac{g-1}{2} \rfloor \geq \frac{g-1}{2} - 1$$

Tehát a mohó algoritmus  $O(n^{1 + \frac{2}{t-1}})$  élű spannert konstruál. Ezzel a spannerünk ritkaságát is vizsgáltuk.

**2.1.3. Állítás ([ADDJS]).**  *$C$  egy kör  $G'$ -ben,  $e$  pedig egy éle ennek a körnek. Ekkor  $W(C - \{e\}) > tw(e)$ .*

**Bizonyítás.** Ezt is lássuk be indirekt módon. Tegyük fel, hogy van olyan  $C$  kör, amelynek  $e$  éle megszegi az állítás feltételét. Akkor az utolsóként beválasztott  $uv$  élre is megszegi, hiszen ennek súlya nagyobb vagy egyenlő, mint  $e$  súlya (mint ahogy az előző bizonyításban is kihasználtuk ezt):

$$W(C - uv) \leq W(C - \{e\}) \leq tw(e) \leq tw(uv).$$

Az egyenlőtlenségsorozat elejét és végét nézve látszik, hogy elhagyva  $uv$  élet megint kapunk egy olyan utat  $u$  és  $v$  között melynek súlya  $> tw(uv)$ , ezzel megvan az ellentmondás.  $\square$

Az algoritmus tekinthető a Kruskal algoritmus általánosításának, amely minimális feszítőfát keres egy gráfban. Mindkettő mohó módon, egyesével növeli a keresett gráf éleit, és hasonlóan egyszerű feltételt ellenőriz minden egyes lépésben. A következő állításhoz felhasználjuk Kruskal algoritmusát is, mivel azzal tudjuk előállítani például az  $MST(G)$ -t:

**2.1.4. Állítás ([ADDJS]).**  *$MST(G)$ -t tartalmazza az algoritmus által létrehozott  $G'$  spanner.*

**Bizonyítás.** Az állításunk bizonyításához egy annál erősebbet fogunk belátni. Legyenek a jelölések a következők:  $\emptyset = G'_0, G'_1, \dots, G'_{S(E)} = G'$  egy sorozat, amely reprezentálja  $G'$  növekedését az algoritmus során. Tehát  $G'_i$   $G$ -nek azon részgráfját jelöli, melyet az  $i$ -edik él hozzáadása után kapunk. Minden fázisban az aktuális  $G'_i$  egy olyan részgráf, amely különböző komponensekből áll, és a végén fog egy teljesen összefüggő gráffá kialakulni, tehát egy nagy komponenssé. Most nézzük a Kruskal algoritmust, amiben a mohó spanneres algoritmushoz hasonlóan az éleket súlyuk szerint nem csökkenő sorban vizsgáljuk. Itt  $\emptyset = M_0, M_1, \dots, M_{S(E)} = MST(G)$  reprezentálja az  $MST(G)$  növekedését,  $M_i$  pedig az  $i$ -edik él hozzáadása után kapott fát. Ezt az élet akkor adjuk hozzá, ha az eddig létrehozott fánkban nem képez kört az eddigi élekkel. Itt minden egyes lépésben egy erdők van, amely a végén egy fává fog fejlődni.

Az erősebb állítás:

**2.1.5. Állítás ([ADDJS]).** *Minden  $i$ -re a komponensek száma  $M_i$ -ben és  $G'_i$ -ben megegyezik. Valamint minden egyes komponensét  $M_i$ -nek  $G'_i$  a hozzá tartozó komponense tartalmazza.*

**Bizonyítás.** Képzeljük úgy, hogy a két algoritmus párhuzamosan fut a  $G$  gráfon. Mindkettő, a fa és a spanner is  $n$  darab komponenssel kezd, tehát a nulladik lépésben igaz az állítás. Teljes indukcióhoz tegyük fel, hogy igaz az  $i$ . lépésig, és az  $i + 1$ -edikre akarjuk látni, hogy itt is teljesül. A következő él a sorrendben  $e = uv$ .

*1.eset:*

$u$  és  $v$  ugyanabban a komponensben vannak  $M_i$ -ben. Ekkor az  $uv$  él egy kört alkotna ebben a komponensben, ergo a Kruskal algoritmus nem fogja bevenni. Az indukciós feltevés szerint ezek  $G'_i$ -ben is egy komponensbe tartoznak. Ha  $e$  nincs benne  $G'_{i+1}$ -ben a mohó algoritmus lépése után, lényegében semmi nem történik, és az indukciós feltevés igaz maradt. Ha pedig  $e$  része lesz  $G'_{i+1}$ -nek. akkor  $G'_i$ -nek ugyanazon komponenséhez lesz hozzáadva, és a feltevés igaz marad.



*2.eset:*

$u$  és  $v$  különböző komponensben vannak, azaz a Kruskal beveszi  $MST(G)$ -be. Tehát  $M_i$  két komponense egy komponenssé válik  $M_{i+1}$ -ben. Az indukciós feltevés szerint  $u$  és  $v$   $G'_i$ -ben is különböző komponensben voltak. Nincsenek összekötve, a mohó algoritmus be fogja venni  $e$ -t, és ez a két komponens  $G'_{i+1}$ -ben is egy lesz már, miközben a feltevés igaz maradt.  $\square$

Végül pedig  $M_{S(E)} = MST(G)$  és  $G'_{S(E)} = G'$ , az eredeti állításunkat is beláttuk.  $\square$

Ami pedig a futási időt illeti: az algoritmus az összes élen végigmegy, ez  $\binom{n}{2}$  csúcspárt jelent, ami  $O(n^2)$  lépés, Minden ilyen lépésben keres egy legrövidebb utat, ami Dijkstra algoritmussal  $O(n \log n)$  időben megvan. Összességében ez  $O(n^3 \log n)$ . Léteznek ennél hatékonyabb mohó algoritmusok is, melyek  $O(n^2 \log n)$  időben készítenek el egy spannert ([BCFMS]), de ezekkel most nem foglalkozunk részletesen. A fent bemutatott algoritmus talán a legkönnyebben érthető, legegyszerűbb verziója a spannert generáló algoritmusoknak.

## 2.2. Yao-, $\Theta$ -gráf

Következő két algoritmusunk futásának alapja geometriai feltétel lesz, és ebben az esetben úgynevezett geometriai gráfokban fogunk spannert felépíteni. Itt az élek súlya a végpontjainak euklideszi távolsága lesz, ezért nevezzük őket geometriai gráfnak. Ekkor a legrövidebb út két pont között az egyenes, tehát  $d_G(u, v) = w(u, v)$  tetszőleges  $u, v$ -re, és itt  $w(u, v)$  pedig az euklideszi távolság. Most a két dimenziós esetet vizsgáljuk.

Legyen  $V$  csúcsok egy halmaza a síkon,  $k$  pedig egy egész szám, amely  $\geq 4$ . Mindkét algoritmusban végigmegyünk az összes  $v \in V$  csúcson,  $v$  körül a síkot felosztjuk  $k$  darab diszjunkt kúpra, melyeket két-két  $v$ -ből kiinduló félegyenes határol, ezek lesznek az alkotók, a szögük pedig  $\Theta = 2\pi/k$ . (Itt az is lehetséges, hogy a szöget adjuk meg inputként, és utána abból kapjuk a  $k$  paramétert  $k = 2\pi/\Theta$  segítségével. Az eredmény ugyanaz lesz, csak eldönthető, hogy mivel akarjuk megadni az inputot.) Ezeket az alkotókat  $r_j$ -vel jelöljük. A  $j$ -edik kúpot  $r_j$  és  $r_{j+1}$  fogja határolni, kivéve persze az utolsót. A számozást pozitív irányban végezzük el. A  $v$  csúcsot minden egyes kúpból pontosan egy darab másikkal kötjük össze, és ebben fog a két algoritmus eltérni egymástól, hogy pontosan melyikkel. A Yao-gráfban (jelöljük ezt  $Y_k$ -val) a  $v$ -hez legközelebbi csúcsot választjuk, a  $\Theta$ -gráfban (ez legyen  $\Theta_k$ ) pedig azzal a csúccsal húzunk be élet, melynek  $r_j$ -re vett merőleges vetülete legközelebb van  $v$ -hez. Tehát minden egyes csúcsot  $v$  kivételével merőlegesen vetítünk annak a kúpnak a negatív irányban elérhető alkotójára, amiben benne vannak. Ezzel kapunk egy  $t$ -spannert  $t = 1 + O(\frac{1}{k})$ -re,  $O(k)$  éllel. Ezt a  $\Theta$ -gráfban fogjuk most belátni  $k \geq 5$  esetben.

Ennek bizonyításához egy ennél bonyolultabb esetet fogunk vizsgálni, amikor is úgynevezett rendezett  $\Theta$ -gráfokat nézünk. Ez annyiban különbözik a fent bevezetettől, hogy a gráf csúcsait rendezzük  $\pi$  (permutáció) szerint. Ekkor néhány új jelölés:  $\pi_v$  a  $v$  csúcs rendje  $\pi$  szerint,

$$P_v = \{u \in V : \pi_u < \pi_v\}, S_v = V \setminus (\{v\}).$$

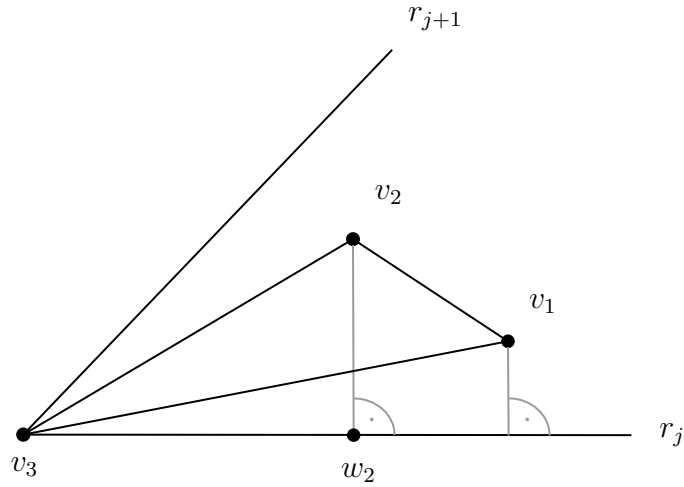
A csúcsok közül pedig csak  $P_v$  elemeit vetítjük merőlegesen, és ezek közül választjuk ki kúponként azt az egyet, melyet összekötünk  $v$ -vel.

**2.2.1. Állítás ([BGM]).** *Tetszőleges  $V$ ,  $\pi$  választása mellett a  $\pi$ -rendezett  $\Theta$ -gráf  $t$ -spanner lesz legfeljebb  $kn$  éllel, ahol  $t = \frac{1}{\cos(\Theta) - \sin(\Theta)}$ .*

**Bizonyítás.** Az állításunk egyik fele egyértelmű,  $k$  db kúp van minden  $v$ -bez, és minden

$v$ -hez minden kúpból pontosan egy darab szomszédot választunk  $\Rightarrow kn$  darab élünk lesz legfeljebb a végén.

A második felét  $\pi$ -szerinti indukcióval látjuk be. Legyenek  $v_1, v_2, v_3$  csúcsok, ahol  $\pi_{v_1} < \pi_{v_2} < \pi_{v_3}$ , és éppen a  $v_3$  csúcsot vizsgáljuk, a másik kettő azonos kúpban van. Továbbá legyen  $w_2$  a  $v_2$  csúcs merőleges vetülete  $r_j$ -re, és tegyük fel hogy  $w_2$  a  $v_3$ -hoz legközelebbi  $r_j$ -n. Feltehető az is, hogy a  $v_3v_2$  szakasz és  $r_j$  között található csúcs a  $v_1$ .



Ekkor:

$$d_{G'}(v_3, v_1) \leq d_{G'}(v_3, v_2) + d_{G'}(v_1, v_2) \leq w(v_3, v_2) + tw(v_2, v_1).$$

Itt az első egy háromszög-egyenlőtlenség, a másodikat pedig úgy kapjuk, hogy  $d_{G'}(v_3, v_2) = w(v_3, v_2)$ , mert  $v_2$  volt a legközelebbi, ergo azt az élet vettük be a spannerbe. A  $d_{G'}(v_1, v_2) \leq tw(v_2, v_1)$  egyenlőtlenség pedig az indukciós feltevésből jön. Mivel  $v_1$  nincs elválasztva  $r_j$ -től a  $v_3v_2$  szakasszal, ezért

$$w(v_2, v_1) + w(v_3, w_2) \leq w(v_2, w_2) + w(v_3, v_1)$$

$\Downarrow$

$$w(v_2, v_1) \leq w(v_2, w_2) + w(v_3, v_1) - w(v_3, w_2)t$$

Ezzel folytatjuk az előző egyenlőségsorozatot:

$$w(v_3, v_2) + tw(v_2, v_1) \leq w(v_3, v_2) + tw(v_2, w_2) + tw(v_3, v_1) - tw(v_3, w_2) \leq tw(v_3, v_1)$$

Itt az utolsó lépéshez tekintsük a  $v_3w_2v_2$  derékszögű háromszöget:

A  $v_3$ -nál lévő szöget jelölje  $\alpha$ , ergo a  $|w_2v_2| = \sin(\alpha)$  valamint  $|v_3w_2| = \cos(\alpha)$ , ha felteesszük, hogy az átmérő 1 hosszú. Tudjuk, hogy  $\alpha \leq \Theta$ . A kérdéses egyenlőtlenségbe behelyettesítve:

$$1 + t \sin(\alpha) - t \cos(\alpha) + tw(v_1, v_3) \leq tw(v_1, v_3)$$

Mivel  $t = \frac{1}{\cos(\Theta) - \sin(\Theta)}$ :

$$\frac{\cos(\Theta) - \sin(\Theta) + \sin(\alpha) - \cos(\alpha)}{\cos(\Theta) - \sin(\Theta)} + tw(v_1, v_3) \leq tw(v_1, v_3) = td_G(v_1, v_3)$$

És mivel  $[0, \pi/2]$ -n van minden szög, ezért  $\cos(\Theta) \leq \cos(\alpha)$  és  $\sin(\alpha) \leq \sin(\Theta)$ ,  $\cos(\Theta) - \sin(\Theta) \leq \sin(\alpha) - \cos(\alpha) \Rightarrow$  a tört negatív vagy 0 a bal oldalon, tehát az egyenlőtlenség igaz.

Ebből kapjuk, hogy  $d_{G'}(v_3, v_1) \leq w(v_3, v_1) = td_G(v_3, v_1)$ , tehát kész vagyunk az indukciós lépésünkkel, ezzel az állítást is bizonyítottuk.  $\square$

## 2.3. WSPD

(Well-separated pair decomposition)

Ezen algoritmusához is síkbeli pontokat fogunk vizsgálni, legyen  $V$  egy  $n$  pontot tartalmazó halmaz, ezek lesznek a csúcsok, a súlyok megint euklideszi távolságok lesznek. Egyszerűsítsük úgy a gráfot, hogy a legkisebb távolság két pont között legyen egység hosszú, a legnagyobb pedig:

$$\alpha := \max_{e \in E} w(e)$$

Vezessük be a következő jelölést a csúcsok csökkenő sorozatára:  $V = V_0 \supset V_1 \supset V_2 \supset \dots \supset V_h$  (ezek lesznek az úgynevezett szintek,  $h$  pedig a szintek számát jelöli), ahol minden  $i > 0$ -ra  $V_i$ -t  $V_{i-1}$ -ből kapjuk úgy, hogy maximális részhalmazzt keresünk  $V_{i-1}$ -ben, hogy ha a részhalmaz, azaz  $V_i$  minden eleme köré egy  $2^i$  sugarú kört rajzolva, a körök lefedik  $V_{i-1}$  összes elemét. Valamint a körök középpontjai legalább  $2^i$  távolságra vannak egymástól, tehát egyik kör sem fedi le a másiknak a középpontját. A minimális távolság  $V_i$ -ben így több mint  $2^i$  lesz, ezt a mennyiséget, az  $i$ -edik szint sugarának hívjuk. Ez egyfajta hierarchikus rendszert ad a pontok között. Ezeket a szinteket az eredeti  $V$  halmazból kiindulva tudjuk meghatározni, az egyes szinteken lévő pontokat mohó módon választjuk meg.

**2.3.1. Állítás ([GGN]).** *A hierarchiában legfeljebb  $\lceil \log_2 \alpha \rceil$  szint lehetséges, azaz  $h \leq \lceil \log_2 \alpha \rceil$ .*

**Bizonyítás.** Legyen  $v \in V_i$ , ezen szint sugara  $2^i$ . Nézzük azt az  $i$ -t, amire már  $\alpha \leq 2^i$  teljesül. Ekkor  $v$  köré egy  $2^i$  sugarú kört rajzolva az összes pontot lefedjük a gráfban. Az egyenlőtlenség logaritmusát véve a bizonyítandó állítás jön ki.  $\square$

Hogy spannert kapjunk, meg kell adnunk, hogy milyen éleket húzunk be a  $G'$ -be. Kétféle élünk lesz: az egyik fajta azonos szinten lévő csúcsokat köt össze. Egy  $u, v \in V_i$  csúcspár akkor legyen összekötve, ha  $d(u, v) \leq c2^i$  teljesül, ahol

$$c = 4 + 16/\varepsilon$$

valamilyen  $\varepsilon > 0$  választásával. Tehát itt ez az  $\varepsilon$  lesz a változtatható paraméter az algoritmusban a  $V$  csúcshalmaz mellett, melyet szintén mi adunk meg inputként. Valamint

minden  $u \in V_{i-1} \setminus V_i$  csúcsot összekötünk  $V_i$  belüli  $v$  csúcsokkal, amelyre  $d(u, v) \leq 2^i$ . Elnevezés  $v$ -re: az  $u$  csúcs szülője. Minden csúcsnak pontosan egy szülője lesz, de lehet több gyereke is. Ergo, ezeket az éleket könnyen tudjuk egy fával szemléltetni. Nyilván az egész gráfunk nem lesz fa, mert vannak a szinteken is élek, de a szülő-gyerek kapcsolatot egy fa írja le.

Fontos megjegyeznünk, hogy az algoritmus nem határoz meg egyértelműen egy konkrét gráfot, hiszen a  $V_i$  szintek választására több lehetőség is van.

A későbbi bizonyításokhoz vezessünk még be néhány jelölést: jelezhetjük, hogy a  $v$  csúcs az  $i$ -edik szinten van:  $v^{(i)}$ . A  $v^{(i-1)}$  csúcsnak egy szülője van az  $i$ -edik szinten, ez legyen  $P(v^{(i-1)})$ . Ezt lehet úgy is használni, hogy a  $v$  csúcs szülőinek halmaza  $P(v)$ , ahol az összes szinten végighaladva a szülők egy láncát kapjuk. Valamint  $P^{(i)}(v)$  legyen a  $v$  csúcs őse az  $i$ -edik szinten, ahol ugye  $v \in V = V_0$ . Ezzel a jelöléssel egy tetszőleges csúcsból kiinduló láncot kapunk a csúcsok szülőit keresve egyre magasabb szinteken:  $vP^{(1)}(v)P^{(2)}(v)\dots P^{(i)}(v)$ .

**2.3.2. Állítás.** *[[GGN]] Egy ilyen láncnak a hossza legfeljebb  $2^{i+1}$ , valamint a  $v$  csúcs  $i$ -edik szinten lévő szülője  $P^{(i)}(v)$  a  $v$ -től legfeljebb  $2^{i+1}$  távolságra van (euklideszi távolságot használva).*

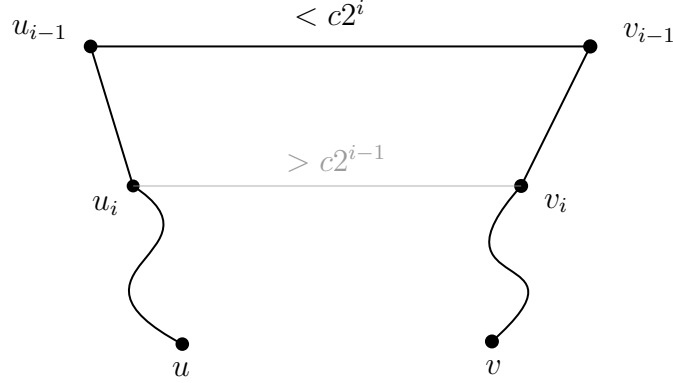
**Bizonyítás.** Az élek bevezetésénél láttuk, hogy szülő és gyerek távolsága kevesebb mint  $2^i$  (az  $i$ -edik és  $(i-1)$ -edik szintek között), tehát egy ilyen lánc hossza legfeljebb  $2+2^2+\dots+2^i$ , aminek felsőbecslése a  $2^{i+1}$ . Háromszög-egyenlőtlenséget használva pedig a bizonyítás második felével is kész vagyunk.  $\square$

Az előbbi jelölést „visszafelé” is bevezethetjük:  $C_{i-1}(v)$  az  $i$ -edik szinten lévő  $v^{(i)}$  csúcs gyerekeit foglalja össze. A  $v$  csúcs szomszédai az  $i$ -edik szinten pedig legyenek  $N_i(v)$ . Ezzel pedig már készen állunk belátni, hogy az algoritmusunk tényleg egy spannert ad nekünk.

**2.3.3. Állítás ([GGN]).** *Ezen élekkel és az eredeti csúcshalmazzal egy  $(1 + \varepsilon)$ -spannert kapunk. ( $\varepsilon$ -t az élek definiálásánál vezettük be.)*

**Bizonyítás.** Vegyünk tetszőleges  $u, v \in V_0$  csúcsokat, és keressük meg a legkisebb  $i$ -t, amire  $V_i$ -n az őseik,  $P^{(i)}(u)$  és  $P^{(i)}(v)$  össze vannak kötve egy éllel. Tehát távolságuk legfeljebb  $c2^i$ , és mivel a legkisebbet kerestük, az eggyel kisebb szinten nincsenek összekötve a szülők, azaz ott a távolság több mint  $c2^{i-1}$ . Legyen  $u_i = P^{(i)}(u)$ ,  $v_i = P^{(i)}(v)$ ,

$u_{i-1} = P^{(i-1)}(u)$  és  $v_{i-1} = P^{(i-1)}(v)$ . Ezzel a jelöléssel az előző megállapítás:  $u_i$  és  $v_i$  távolsága több mint  $c2^{i-1}$ . Nézzük a következő utat  $u$  és  $v$  között:  $u$  és  $u_i$  között a korábban említett lánc, amely az  $u$  szülőin megy végig, az él  $u_i$  és  $v_i$  között, majd a lánc, ami  $v_i$ -t és  $v$ -t köti össze.



Jelöljük ezt  $\Lambda(u, v)$ -vel. Ez az út csak gráfbeli élekből áll, és ha belátjuk, hogy hossza legfeljebb  $(1 + \varepsilon)w(u, v)$ , akkor kész vagyunk a bizonyítással. A 2.3.2 állítás miatt  $d_{G'}(u, u_{i-1}) \leq 2^i$  és  $d_{G'}(v, v_{i-1}) \leq 2^i$ . Az élek definíciója miatt  $w(u_i, v_i) \leq c2^i$  és mivel  $u_{i-1}$  és  $v_{i-1}$  nincsenek összekötve  $d_{G'}(u_{i-1}, v_{i-1}) > c2^{i-1}$ . Ezeket helyettesítsük be a következőbe:

$$\begin{aligned} w(u, v) &\geq d_{G'}(u_{i-1}, v_{i-1}) - d_{G'}(u, u_{i-1}) - d_{G'}(v, v_{i-1}) \\ d_{G'}(u, v) &> (c - 4)2^{i-1} \\ &\Downarrow \\ 2^{i-1} &< \frac{d_{G'}(u, v)}{c - 4} \end{aligned}$$

Kaptunk egy alsó becslést a két pontunk távolságára. Az eddig vizsgált utunkra pedig a következő becsléseket írhatjuk fel:

$$\begin{aligned} W(\Lambda(u, v)) &\leq d_{G'}(u, u_i) + w(u_i, v_i) + d_{G'}(v, v_i) \\ W(\Lambda(u, v)) &\leq 2 \cdot 2^i + w(u_i, v_i) \leq w(u, v) + 8 \cdot 2^{i-1} \end{aligned}$$

Felhasználva a fentit és  $\varepsilon$  definícióját:

$$w(u, v) + 8 \cdot 2^{i-1} \leq (1 + 16/(c - 4))w(u, v) = (1 + \varepsilon)w(u, v)$$

$\Rightarrow$  Az algoritmus által generált gráf egy  $(1 + \varepsilon)$ -spanner lesz.  $\square$

Az is belátható, hogy az így kapott spannernek  $O(n/\varepsilon^d)$  éle lesz, ahol  $d$  azt jelzi, hogy milyen dimenziós térben vesszük  $V$ -t. De mivel most a síkban vagyunk, ez  $O(n/\varepsilon^2)$ -et jelent. Ez a gráf ritkasága.

## 2.4. Delaunay-gráfok

Térjünk át a negyedik algoritmusunkra, melyhez először a Delaunay-gráfot fogjuk definiálni, melynek vannak különböző általánosításai. Mindegyik verziójára belátható, hogy spanner egy adott konstansra. Ezeket fogjuk megvizsgálni most részletesebben.

A Delaunay-gráfot bevezethetjük a Voronoi-diagram duálisaként. Először nézzük, hogy mi is az a **Voronoi-diagram**!

**2.4.1. Definíció.** Vegyünk egy  $V$  csúcshalmazt a síkon, és minden egyes csúcshoz megadunk egy síkrészt olyan módon, hogy az adott  $v$  csúcshoz tartozó síkrészben egy tetszőleges  $p$  pont a  $V$  elemei közül  $v$ -hez van a legközelebb.

Mintha azt akarnánk modellezni, hogy emberek egy városban melyik boltba menjenek vásárolni, ha mindenki a hozzá legközelebb lévő boltba szeretne eljutni a leggyorsabban. Legyenek  $V$  elemei a boltok, az emberek pedig a sík pontjaiban lakhatnak. Ekkor a Voronoi-diagram alapján el lehet dönteni a kérdést, mert mindenkire az a bolt van legközelebb, amelyikhez tartozó síkrészben lakik. A Delaunay-gráfhoz pedig ugyanezt a  $V$  csúcshalmazt használjuk, a következőképpen:

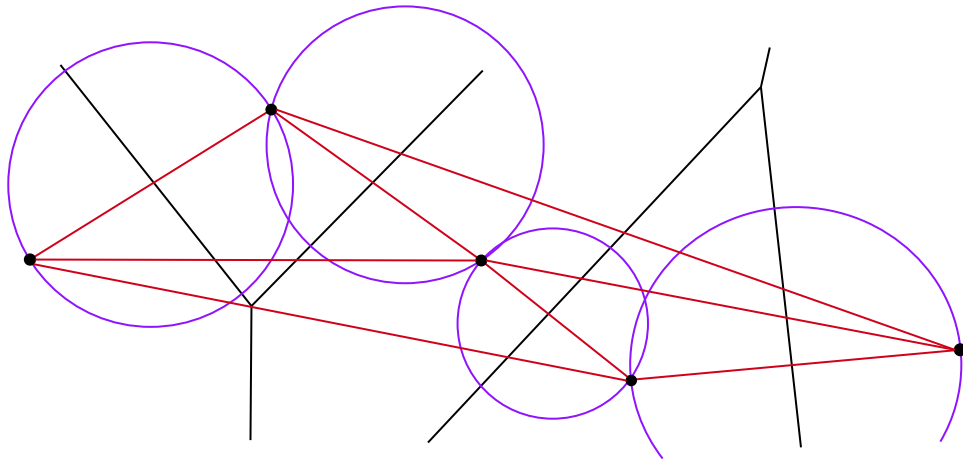
**2.4.2. Definíció.** Adott egy  $V$  csúcshalmaz a síkon, és annak a Voronoi-diagramja. Ekkor a  $V$  csúcsokhoz tartozó **Delaunay-gráfot** úgy kapjuk, hogy a csúcsok közül azokat kötjük össze éllel, melyeknek van közös határuk a Voronoi-diagramban.

Ezt a gráfot Delaunay-háromszögelésnek is nevezhetjük, hiszen a gráfunk „belseje” háromszögekből fog állni, és az is könnyen látható, hogy ez egy síkbarajzolható gráf.

Ezzel egy ekvivalens definíció a Delaunay-gráfra:

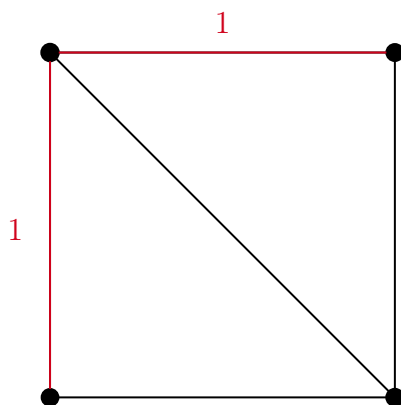
**2.4.3. Definíció.** Adott  $V$  csúcshalmazban azokat a pontokat kötjük össze, melyekhez létezik egy kör, amely csak az adott két pontot tartalmazza  $V$  elemei közül, így kapjuk a  $V$ -hez tartozó **Delaunay-gráfot**.





Ezen az ábrán jól látszik a két ekvivalens definíció néhány pontra. A fekete félegyenesek a Voronoi-diagram határai, a piros szakaszok pedig az élek, melyeket behúzzhatunk a körök vagy a diagram alapján. (Nem minden élhez tartozó kört jelenít meg az ábra.)

Most pedig vizsgáljuk meg, hogy egy Delaunay-gráf spanner lesz-e, és ha igen, akkor milyen  $t$  értékre? A Delaunay-gráfok geometriai síkgráfok, ez könnyen végiggondolható. Most pedig belátjuk, hogy semmilyen geometriai síkgráfra nem tudunk ilyen  $(1 + \varepsilon)$ -spannereket készíteni tetszőleges  $\varepsilon$ -ra. Legyen négy pontunk egy egységnyezet csúcaiban, a normánk legyen az euklideszi norma a síkon. Ekkor nem húzhatjuk be az összes élet a négyzetben, az egyik átlót ki kell hagynunk. A kihagyott átló hossza  $\sqrt{2}$ , a két pont közötti úté  $G'$ -n belül pedig 2. Tehát  $t \geq \sqrt{2}$ . Tehát ha a spanner egy síkbarajzolható, akkor  $t \geq 2$ .



Jelölések pontosítása a mostani bizonyításainkhoz:  $V \in \mathbb{R}^2$  az előre megadott csúcshalmaz, amelynek teljes gráfja legyen a vizsgált gráfunk:

$$G := K_{|V|}$$

$G'$  pedig legyen a  $V$ -hez tartozó Deluanay-gráf.  $G$ -nek egy spannereként fogjuk vizsgálni  $G'$ -t.

A következő tételt, amely a spannerséget bizonyítja, és a hozzá tartozó segédállítást Keil és Gutwin látták be ([KG]). Először kezdjük az állítás bizonyításával, mert ezt felhasználjuk majd a tételünk bizonyításában.

**2.4.4. Állítás (Keil, Gutwin [KG]).** *Legyen  $u, v \in V$ , tegyük fel, hogy  $L = pq$  egy vízszintes szakasz,  $C$  pedig egy kör, melynek köríve tartalmazza  $u, v$  csúcsokat, és  $L$  alatt nincsen más csúcsa gráfnak. Ekkor  $d_{G'}(u, v) \leq r\Theta$ , ahol a kör sugara  $r$ ,  $\Theta := \angle uOv$  egy középponti szög. (Az egyenlőtlenség jobb oldalán az  $uv$  ív hossza van, azzal becsüljük felül az  $u$  és  $v$  távolságát.)*

**Bizonyítás.** Mindamellett, hogy feltettük,  $L$  egy vízszintes szakasz, hasonlóan az is feltehető, hogy  $u$  balra helyezkedik el  $v$ -hez képest, mivel szerepük felcserélhető, tetszőleges pontok voltak.

Ha  $C$  belsejében semmilyen  $V$ -beli pont nincs, akkor a Deluanay-gráf definíciójából adódóan összekötjük őket egy éllel:  $d_{G'}(u, v) = w(u, v) \leq r\Theta$  minden  $u$  és  $v$ -re írt körre, azaz ebben az esetben kész vagyunk.

A másik esetet  $\Theta$  méretére vonatkozó indukcióval fogjuk belátni. Ekkor van másik  $t \in V$  pont a  $C$  belsejében (nyilván  $L$  fölött). De előtte néhány új objektumot és jelölést vezetünk be a bizonyításhoz. Legyenek  $C_1$  és  $C_2$  körök  $O_1, O_2$  középpontokkal,  $r_1, r_2$  sugarakkal, hogy

$$u, t \in C_1; v, t \in C_2$$

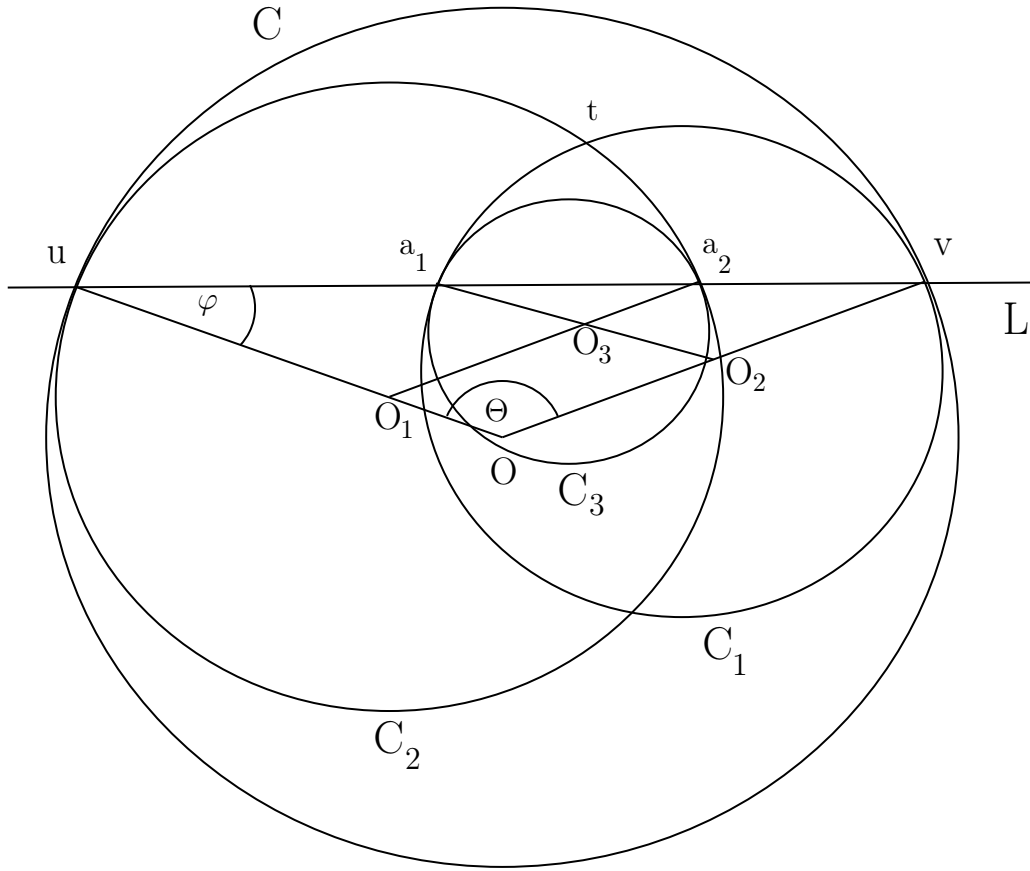
$$O_1 \in uO; O_2 \in Ov$$

és  $C_1$  belülről  $u$ -ban érinti  $C$ -t, hasonlóan  $C_2$  pedig  $v$ -ben.

Legyenek:

$$a_1 := C_1 \cap L; a_2 := C_2 \cap L$$

Az  $a_1$  és  $a_2$  pontok köré írjunk egy kört, hogy a középpontja  $O_3 := a_2O_2 \cap a_1O_1$  legyen, jelöljük  $C_3$ -mal, sugarát pedig  $r_3$ -mal. A  $D$  kör pedig legyen  $u, v, t$  pontok körülírt köre.



Megfigyelések az ábráról, melyeket most nem látunk be részletesen:

(1.) Nincsen csúcsa a gráfnak  $C_1$ -ben az  $ut$  egyenes alatt valamint  $C_2$ -ben a  $tv$  egyenes alatt.

(2.) Az  $uOv$ ,  $uO_1a_1$ ,  $a_2O_2v$ ,  $a_2O_3a_1$  háromszögek hasonlóak egymáshoz és egyenlő szárúak. Alapon lévő szögüket jelölje  $\varphi$ . (A hasonlóság a szögekből könnyen látszik.)

(3.)  $\cos \varphi = \frac{w(u,a_1)}{2r_1} = \frac{w(v,a_2)}{2r_2} = \frac{w(a_2,a_1)}{2r_3} = \frac{w(u,v)}{2r}$ , ahol ugye  $w$  volt a lehetséges élek súlya, ami esetünkben az euklideszi távolságuk a síkban.

A bizonyítás első lépéseként transzformáljuk a  $C$  kört úgy, hogy a köríven  $u$  és  $v$  mellett egy  $z \in V$  pont is rajta legyen, és  $z$   $L$  alatt helyezkedik el, de a köríven, nem a kör belsejében. Miközben az a tulajdonság megmarad, hogy nincsen másik csúcs  $C$  belsejében  $L$  alatt. Mivel ezzel a  $\Theta$  szög csak csökkent vagy nem változott, az  $r\Theta$  érték is, tehát, ha az állítást erre a transzformált körre belátjuk, akkor az eredetire is igaz lesz. Ha nincs ilyen

$z$  pont, akkor az  $u$  és  $v$  csúcsok újfent össze vannak kötve éllel, tehát az állítás igaz.

Erre a transzformált  $C$  körre pedig a  $\Theta$  méretére vonatkozó indukcióval fogjuk belátni a kívánt becslést. Legyen  $\Theta$  a lehető legkisebb érték, akkor ha  $C$  nem üres, és a fentiek alapján definiáltuk  $C_1$ -et, akkor a (2.)-t felhasználva:  $\Theta_1 := uO_1t\angle < uO_1a_1\angle = \Theta$ , de ez ellentmond annak, hogy  $C$  a legkisebb szöggel rendelkező ilyen kör, azaz  $C$  üres, tehát  $uv$  egy éle a gráfnak, igaz az állítás.

Az indukciós lépéshez vegyünk egy konkrét  $C$  kört és hozzá tartozó  $\Theta$  értéket, és tegyük fel, hogy az ennél kisebb szöggel rendelkező körökre az állítás igaz.

Tegyük fel, hogy  $a_2$  az  $a_1$  bal oldalán található (a másik esetet is hasonlóan tudjuk végigvenni). A (2.) megállapításunk alapján:  $\Theta_1 = uO_1t\angle < uO_1a_1\angle = \Theta$  valamint  $\Theta_2 := tO_2v\angle < a_2O_2v\angle = \Theta$ , és figyelembe vesszük (1.)-t is, akkor a  $C_1$  és  $C_2$  körökre használhatjuk az indukciós feltevést:

$$d_{G'}(u, v) \leq d_{G'}(u, t) + d_{G'}(t, v) \leq r_1\Theta_1 + r_2\Theta_2$$

$$\begin{aligned} r_1\Theta_1 + r_2\Theta_2 &= r_1(\Theta - (\Theta - \Theta_1)) + r_2(\Theta - (\Theta - \Theta_2)) = r_1\Theta + r_2\Theta - [r_1(\Theta - \Theta_1) + r_2(\Theta - \Theta_2)] \\ &< r_1\Theta + r_2\Theta - r_3\Theta = (r_1 + r_2 - r_3)\Theta \end{aligned}$$

Ez utolsónál felhasználtuk a következő egyenlőtlenséget:

$$r_3\Theta + w(a_2, a_1) < r_1(\Theta - \Theta_1) + r_2(\Theta - \Theta_2) + w(a_2, a_1),$$

ahol a bal oldalon a  $C_3$   $L$  alatti konvex alakzat határvonalának hossza van, jobb oldalon pedig a  $C_1 \cap C_2$   $L$  alatti konvex alakzatáé. Az egyenlőtlenségünkben pedig ott tartottunk, hogy:

$$d_{G'}(u, v) \leq (r_1 + r_2 - r_3)\Theta \stackrel{(3.)}{=} \left( \frac{w(u, a_1)}{2 \cos \varphi} + \frac{w(a_2, v)}{2 \cos \varphi} - \frac{w(a_2, a_1)}{2 \cos \varphi} \right) \Theta = \frac{w(u, v)}{2 \cos \varphi} \Theta = r\Theta$$

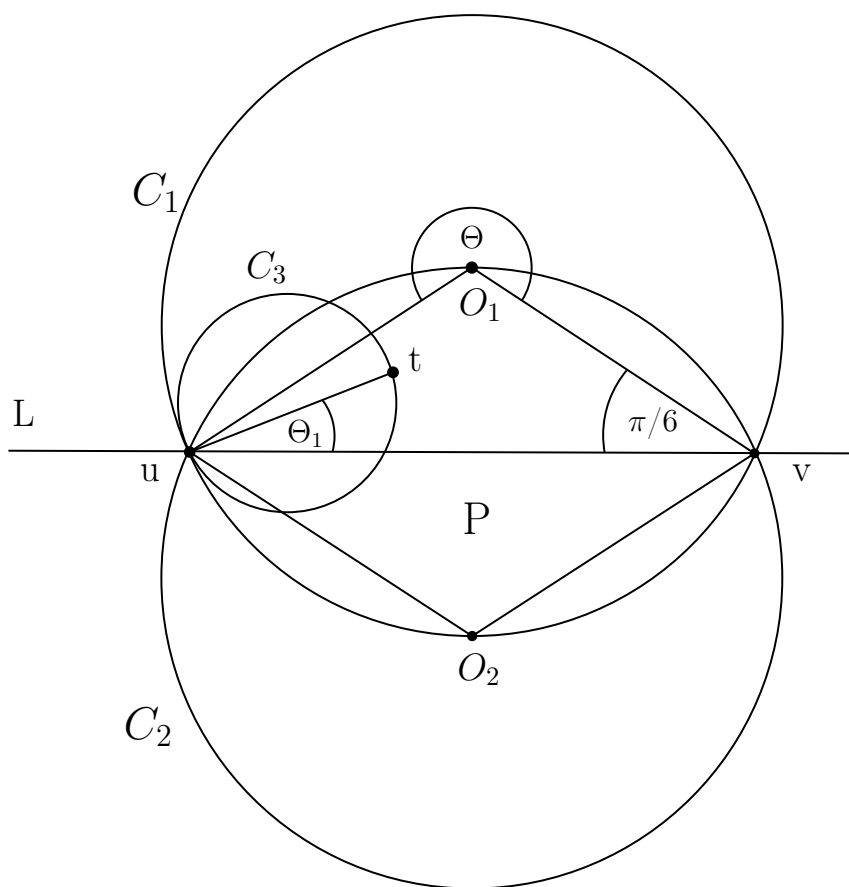
Ha pedig  $a_1$  és  $a_2$  helyzete pont fordítva van, akkor

$$d_{G'}(u, v) \leq r_1\Theta_1 + r_2\Theta_2 \leq r_1\Theta + r_2\Theta \stackrel{(3.)}{=} \left( \frac{w(u, a_1)}{2 \cos \varphi} + \frac{w(a_2, v)}{2 \cos \varphi} \right) \Theta = \frac{w(u, v)}{2 \cos \varphi} \Theta = r\Theta$$

Tehát minden esetben beláttuk az indukciós lépést, kész vagyunk a bizonyítással.  $\square$

**2.4.5. Tétel (Keil, Gutwin [KG]).** *A Deluanay-gráf egy  $\lambda$ -spanner, azaz  $d_{G'}(u, v) \leq \lambda w(u, v)$  tetszőleges  $u, v \in V$ -re, ahol  $\lambda = \frac{2\pi}{3 \cos \pi/6} \approx 2,42\dots$*

**Bizonyítás.** Hasonlóan az előző állításhoz, itt is jelöljük a két tetszőleges vizsgált csúcsot  $u, v$ -vel, összekötő szakaszukat  $L$ -lel. Állítsunk két egyenlő szárú háromszöget  $L$ -re mindkét irányba, ahol az alapon lévő szögek  $\pi/6$  nagyságúak legyenek, a két új csúcs pedig  $O_1$  és  $O_2$ , ahogy az az ábrán is látszik. Írjuk  $u$  és  $v$  köré két kört,  $C_1$ -et és  $C_2$ -t, hogy a középpontjuk a két újonnan felvett,  $O_1, O_2$  pontok legyenek. Jelöljük a két kör belső részeit együtt  $P$ -vel.



A bizonyítást itt is egy indukcióval kezdjük, csak most a  $d_{G'}(u, v)$  szerintivel, ahol ugye ezzel a Deluanay-gráfban lévő minimális út hosszát jelöltük. Ennek a minimális értéke  $w(u, v)$ , amikor  $u$  és  $v$  össze vannak kötve  $G'$ -ben, ekkor tehát igaz az egyenlőtlenségünk. Nézzük az indukciós lépést. Ekkor az első eset az, hogy  $P$  belsejében nincsen másik  $V$ -beli csúcs. Ekkor az előbbi állítás alkalmazható  $C_1$ -re:  $d_{G'}(u, v) \leq r_1 \Theta_1$ , ahol  $r_1$  a sugara,  $\Theta_1$  pedig a „felső”  $uO_1v$  szög, ami  $= 2\pi - (\pi - 2(\pi/6)) = \frac{4}{3}\pi$ ,  $r_1$  pedig  $= \frac{w(u, v)}{2 \cos \pi/6}$ . Tehát:

$$d_{G'}(u, v) \leq \frac{2\pi}{3 \cos \pi/6} w(u, v) = \lambda w(u, v)$$

Másik esetben feltehetjük, hogy van egy pont  $P$ -ben az  $uvO_1$  bal oldalán. Rajzoljunk egy harmadik kört a  $p$  és a másik csúcs,  $t$  köré úgy, hogy középpontja az  $uO_1$ -en legyen. Jelöljük ezt a kört  $C_3$ -mal. Valamint az is feltehető, hogy  $C_3$ -ban nincsen másik csúcs  $ut$  egyenese alatt, azaz alkalmazható erre a körre is az állítás:

$$d_{G'}(u, v) \leq d_{G'}(u, t) + d_{G'}(t, v) \leq r \angle uO_3t + \lambda w(t, v) \leq \lambda w(u, v)$$

Ahol az utolsó egyenlőtlenségrészt egyszerű elemi számításokkal be lehet látni.  $\square$

Ezzel a kérdéssel később többen is foglalkoztak, és megpróbálták ezt a konstans dilatációs tényezőt, amit most  $\lambda$ -val jelöltünk, finomítani. Például Xia, akinek sikerült ezt levinni 1,998-ra ([Xia]). Azonban érdekes módon nem erre a Delaunay-gráfra sikerült bizonyítani először valamilyen  $t$  értékre, hogy spanner, hanem más, nem a klasszikus értelemben vett „körökkel” definiált Delaunay-gráfokra, melyeket most meg is nézünk részletesebben.

### Általános Delaunay-gráfok

Tehát azt keressük, hogy hogyan lehetne a Delaunay-gráfot valahogy általánosítani. Többek között úgy, hogy a szokásos értelemben vett köröket, melyet az euklideszi norma síkbeli egységömbjeként kapunk, más normákból származtatott egységömbökkel vagy esetleg egy tetszőleges konvex objektummal, például egy szabályos háromszöggel helyettesítjük. Ez utóbbihoz tartozó távolságot a következőképp vezethetjük be: vegyünk egy tetszőleges konvex alakzatot a síkon, és jelöljük ki a középpontját. Ez lesz az egységkörünk. Ekkor tetszőleges  $p$  és  $q$  pontok távolságát úgy kapjuk, hogy  $p$  köré írunk egy egységkört, és addig nagyítjuk vagy kicsinyítjük, amíg eléri a  $q$  pontot a kör körvonala. Ekkor a két pont távolsága az az arány, tényező lesz, amivel az egységkört változtattuk. Ezt **konvex távolságfüggvénynek** hívunk, és nem feltétlenül kaptunk vele metrikát. Hiszen a szimmetria tulajdonság csak abban az esetben teljesül, ha az egységkörünk, amivel megadtunk egy adott távolságot, a középpontjára szimmetrikus. Ellenkező esetben  $d(p, q) \neq d(q, p)$  is előfordulhat, mert először  $p$  majd  $q$  köré rajzoltuk az egységkört, és nem ugyanazt az arányt kapjuk. A továbbiakban nem ilyen távolságfüggvényeket fogunk vizsgálni, de a Delaunay-gráfnál nem is a szimmetria tulajdonságot használjuk ki, hanem azt nézzük, hogy adott pontok illeszkednek-e ezekre a körökre?

### Háromszög-Delaunay-gráf

Vizsgáljuk először a szabályos háromszögekből kapott távolságfüggvényünket, és az eb-

ből származtatott Deluanay-gráfot. Az egységkörünkből, azaz a szabályos egységháromszögből kapott távolságfüggvénnyel ugyanúgy tudjuk definiálni a Voronoi-diagramot és a Deluanay-gráfot, hiszen az előbb bevezetett két definíció nem használja ki, hogy pontosan hogyan mérjük a távolságot. Szemléletesen,  $V$  minden pontjából indítunk egy egységkört, mint egy hullámot, ami folyamatosan növekszik, egyszerre az összes pontból, és ahol két hullám találkozik, ott lesz a diagram egy határa. A Deluanay-gráf második definíciója is átvihető a szabályos háromszögekre úgy, hogy akkor lesz él két pont között, ha tudunk úgy rajzolni egy szabályos háromszöget, hogy az adott két pont a határán van valahol, és más  $V$ -beli pont nincs a háromszög, azaz esetünkben kör belsejében. Azaz minden élhez tartozik egy, a végpontjait tartalmazó üres kör. Jelöljük a háromszögekkel, mint egységkörrel megadott Delaunay-gráfot, háromszög-Delaunay-gráfnak. Későbbiekben, ha a körökre hivatkozunk, akkor a szabályos háromszögekre gondolunk. Mindez a definíció formálisan:

**2.4.6. Definíció.** *Adott a síkon egy  $V$  csúcshalmaz. Az ehhez tartozó **háromszög-Deluanay-gráf** pedig egy olyan síkbarajzolható gráf, melynek bármely élének két végpontjára tudunk írni egy kört, melynek belsejében más  $V$ -beli pont nincsen, és mindezen tulajdonságokkal rendelkező gráfok közül is a maximális. (Maximális alatt értjük, hogy tovább nem bővíthető.)*

Lehetséges, hogy egy csúcshalmazhoz több gráf is tartozik (ha van négy pont, ami egy körre esik), de ebben az esetben tetszőlegesen kiválasztjuk az egyiket.

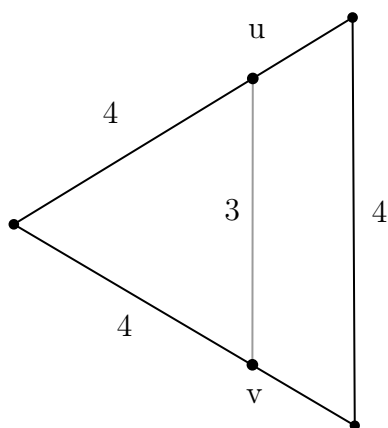
A következő tételek pedig, melyekben belátjuk, hogy milyen spannereket is kaptunk ezekkel a gráfokkal, Chew nevéhez kötődnek.

**2.4.7. Állítás (Chew [Ch89]).** *A háromszög-Delaunay-gráf minden belső háromszögéhez létezik egy kör, amely áthalad a háromszög mindegyik csúcán, és belsejében nincsen másik csúcса a gráfnak.*

Ebben az állításban belső háromszög alatt azokat a gráfbeli háromszögeket értjük, melyek kialakulnak egy háromszög-Delaunay-gráf esetén is. Itt is, ahogy az angol szakirodalomban is Delaunay-háromszögelésnek hívják az ilyen típusú gráfokat.

**2.4.8. Tétel (Chew [Ch89]).** *Legyen  $V$  egy csúcshalmaz a síkon,  $G$  pedig az ehhez tartozó háromszög-Deluanay-gráf. Ekkor  $G$  egy spanner  $t = 2$ -re.*

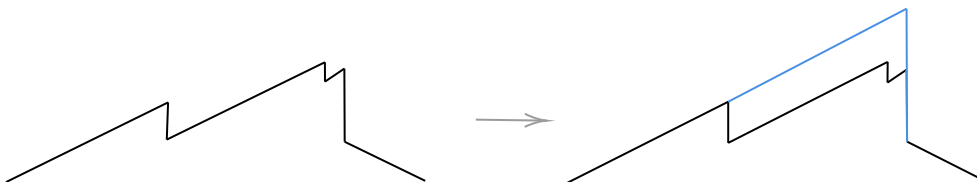
Mutassunk egy példát arra, hogy ez a felső becslés éles  $t$ -re. Vegyünk egy kört, azaz egy szabályos háromszöget, melynek élei 4 egység hosszúak, és  $V$ -nek öt pontja is megtalálható rajta. Ekkor ugye több lehetőség is van élek berajzolására. Az  $u$  és  $v$  csúcsok a háromszög oldalain vannak, és távolságuk 3, a maradék pontok pedig a háromszög csúcsai. Létezik olyan lehetséges élbehúzás, melyben nincsen összekötve  $u$  és  $v$ , és a köztük lévő legrövidebb út hossza 6. Ebből azt kapjuk, hogy  $t \geq 2$  ebben a konkrét gráfban. Tehát nem kapunk ennél kisebb  $t$  értéket az ilyen típusú Delaunay-gráfokra.



A bizonyítást nem nézzük részletesen, de nagyvonalakban szeretnénk végigmenni rajta. Először csak speciális esetben vizsgálunk két csúcsot, amikor egy vízszintes egyenesen vannak. Jelöljük őket  $u$ -val és  $v$ -vel. Egy algoritmussal fogunk utat keresni a Delaunay-gráfban  $u$  és  $v$  között. Az algoritmusban csak azokat az éleket, és a hozzájuk tartozó csúcsokat vizsgáljuk, melyek átmetszik az  $u$  és  $v$ -t összekötő vízszintes szakaszt. Minden ponthoz, amelyek így megmaradtak, berajzoljuk a Delaunay-gráf definíciójában szereplő körülírt üres köröket is. Elindulunk a bal oldalon lévő  $u$  csúcstól ezen körök mentén, amíg a  $v$  csúcsba nem érünk, mindig egy-egy másik csúccsal és éllel bővítve az utunkat. Itt nem a körök körvonalai lesznek az élek, hanem a csúcsokat kötjük össze, melyekhez tartozik az adott köréírható kör, csak a körök segítségével kapjuk meg a csúcsok sorrendjét az úton. Valamint ezekkel a körökkel fogjuk felülbecsülni az út hosszát. Legyen az algoritmus által kapott út  $P$ , a körök mentén lévő út pedig  $Q$ , ami ugye nem gráfbeli éleket használ, de tudjuk róla, hogy  $|P| \leq |Q|$ . Az új út hosszát,  $|Q|$ -t pedig részenként fogjuk becsülni. Aszerint osztjuk részekre  $Q$ -t, hogy mikor metszi a vízszintes szakaszunkat, amely  $u$  és  $v$  közt halad. Egy ilyen részt az út hosszának változtatása nélkül egyszerűen csak „kihajtogatunk”, így egy félszabályos háromszöget kapunk, aminek könnyedén tudjuk



az oldalhosszait meghatározni.



egy útszakasz „kihajtogatva”

Ezek után ez az egész algoritmus, és a becslés is kiterjeszthető egy teljesen tetszőleges  $u$  és  $v$  csúcsra, csak bevezetünk egy  $\alpha$  szöveget, mely az  $uv$  egyenes és az  $u$ -ból induló vízszintes egyenes bezárt szögét jelenti, és  $\alpha$  egyfajta paraméterként jelenik meg a számításainkban. Az előző speciális esetet  $\alpha = 0^\circ$  fedi le.

### $L_1$ -, $L_\infty$ -Delaunay-gráf

Egy másik alternatív Delaunay-gráfot pedig bevezethetünk az  $L_1$  normával, melynek egységköre egy „csúcsára állított” négyzet lesz. Ezt nevezzük  $L_1$ -Delaunay-gráfnak. A háromszögekhez hasonlóan erre az egységkörre és távolságra is könnyen átvihető a definíció a Voronoi-diagramról, azzal a különbséggel, hogy az  $L_1$  metrikát használjuk a távolság méréséhez. A Voronoi-diagramot ebben az esetben, és bármely más metrikával vagy távolsággal definiált esetben is  $O(n \log n)$  időben meg tudjuk konstruálni, ahol  $n = |V|$  ([SH], [Hw], [CD]). Az  $L_1$  esetben pedig  $O(n)$  időben megkaphatjuk az  $L_1$ -Delaunay-gráfunkat a Voronoi-diagramból ([LS]). Összességében akkor ez  $O(n^2 \log n)$  időt jelent. A gráfunk mérete pedig  $O(n)$  lesz ([Ch86]). Ez volt az egyik lehetséges definíció.

Mint a korábbi esetekben, itt is definiálhatjuk a gráfot a Voronoi-diagram nélkül, az egységkörök segítségével, de ebben az esetben kicsit változtatni kell a definíción.

**2.4.9. Definíció.** Adott egy  $V$  csúcshalmaz a síkon, melyhez tartozó  $L_1$ -Delaunay-gráfot a következőképpen kapjuk: minden három csúcsot kössünk össze éllel, amelyek egy háromszöget alkotnak a síkon és:

- (1) létezik a három pont által alkotott háromszöghöz egy körülírható kör, mely alatt az  $L_1$  metrika által definiált kört értjük,
- (2) ebben a körben nincsen másik  $V$ -beli csúcs.

Itt is igaz, úgy mint a háromszögesnél, hogy ha van olyan kör, melyen négy vagy több csúcs van, akkor nem egyértelmű a gráf. Valamint azt is érdekes megfigyelni, hogy a definíció egyik feltétele, hogy lehetséges legyen a három pont köré kört írni. Ez a feltétel  $L_2$ -ben szükségtelen, mivel ott minden háromszög köré írható kör, de esetünkben most ez nem igaz egy tetszőleges háromszögre a síkon. Például ha van két csúcsunk egy vízszintes egyenesen, és a harmadik a kettő között helyezkedik el éppen csak az egyenes fölött.

Az így kapott Delaunay-gráfról is belátható, hogy spanner. Az erre vonatkozó tételt is Chew látta be:

**2.4.10. Tétel (Chew [Ch86]).** *Egy  $V$  csúcshalmaz  $L_1$ -Delaunay-gráfja egy  $\sqrt{10}$ -spanner.*

Ez a tétel nyilván érvényes az  $L_\infty$  normált tér esetén is, hiszen az egységkör ott is egy négyzet, csak más az elhelyezkedésük. De mivel itt nem az egységkör mérete vagy helyzete számít, hiszen csak azt nézzük, hogy két vagy három pont köré írható-e kör, és amúgy is az utak hosszának arányával számoltunk, a két esetben ugyanazt a  $t$  értéket kapjuk.

A bizonyítás lényegileg ugyanúgy működik, mint a háromszögekkel definiált gráfnál. Egy hasonlóan működő algoritmussal készít  $uv$  utat a Delaunay-gráfban, ahol először  $u$  és  $v$  egy horizontális egyenesen vannak, majd kiterjeszti tetszőleges pontokra. Az így kapott utat részekre bontja, és a részeket egyesével becsli. Csak itt nem félszabályos- és szabályos háromszögek határozzák meg az utat, hanem négyzetek, így eltérő értéket fogunk kapni a végén, de a bizonyítás menete, főbb lépései megegyeznek.

Ezt a Chew által kapott értéket is lehet még pontosítani, például Bonichon, Gavoille, Hanusse és Perkovic cikkében olvasható részletesen a következő tétel és annak bizonyítása (mellyel mi most nem foglalkozunk részletesen):

**2.4.11. Tétel ([BGHP]).** *Az  $L_1$ - és  $L_\infty$ -Delaunay-gráfokhoz tartozó nyújtási tényezőre teljesül a következő:  $t \leq \sqrt{4 + 2\sqrt{2}} \approx 2,613\dots$*

Mindemellett érdekes kutatások szólnak arról is, hogy alsó becslést találjunk a Delaunay-gráfok  $t$  értékeihez. Triviálisnak mondható, hogy  $t > \pi/2 \approx 1,5707$ , de ezt azóta belátták 1,5846 ([BDLSV]), majd 1,5932-re is ([XZ]). Ezek az alsó becslések az eredeti,  $L_2$ -Delaunay-gráfra vonatkoznak.

## 3. fejezet

# Összefoglalás, kitekintés

### 3.1. Az algoritmusok összehasonlítása

A négy bemutatott algoritmusunk közül a mohó tér el legjobban a többitől. Ez tetszőleges input gráfra adni fog nekünk egy  $t$ -spannert, tehát itt jelenik meg az, hogy a spanner definíciójában nem feltétlenül szükséges, hogy a kiindulási  $G$ , melyhez hasonlítjuk a leg-rövidebb utakat, egy teljes gráf legyen.

Miközben a három másik geometriai síkgráfokkal foglalkozik, két dimenzióban. Azaz a síkon adott csúcshalmazhoz építenek egy síkbarajzolható gráfot, az élek súlyának a pontok közti euklideszi távolságot tekintik. Ebben a témakörben érthető, hogy ilyen jellegű gráfokkal foglalkozunk, hiszen mint ahogy az elején említett útépitős példában is látszik, a legtöbb hálózatnak, amit modellezni akarunk, fontos kitétele, hogy síkbarajzolható legyen a gráfunk.

Az algoritmusok közötti különbségeket a következő táblázat foglalja össze: 3.1 táblázat.

szempont	mohó	$\Theta$	WSPD	Delaunay
input	$G = (V, E)$ tetszőleges, $t$ paraméter	$k$ paraméter, $V \subset \mathbb{R}^2$ , ( $\Theta := 2\pi/k$ , $n :=  V $ )	$V \subset \mathbb{R}^2$ , $\varepsilon$ paraméter, ( $n :=  V $ )	$V \subset \mathbb{R}^2$ , ( $n :=  V $ )
futási idő	$O(n^2 \log n)$	$O(kn(\log n)^2)$ [BGM]	$O(n \log n)$ [ES]	$O(n^2 \log n)$
$t$	tetszőleges	$1 + O(1/k)$	$(1 + \varepsilon)$	konstans (lásd: 3.2 táblázat)
ritkaság (élek száma)	$O(n^{1+\frac{2}{t-1}})$	$O(kn)$	$O(n/\varepsilon^2)$	$O(n)$
súlyfüggvény	nemnegatív	euklideszi tá- volság	euklideszi tá- volság	euklideszi tá- volság

3.1. táblázat. Algoritmusok összehasonlítása különböző szempontok szerint.

gráf típusa	$t$	cikk
$L_2$ -Delaunay	$4\pi/(3\sqrt{3}) \approx 2,41$	[KG]
$L_2$ -Delaunay	1,998	[Xia]
háromszög-Delaunay	2	[Ch89]
$L_1$ -, $L_\infty$ -Delaunay	$\sqrt{10}$	[Ch86]
$L_1$ -, $L_\infty$ -Delaunay	$\sqrt{4 + 2\sqrt{2}} \approx 2,613$	[BGHP]

3.2. táblázat. Az egyes Delaunay-gráfokhoz tartozó  $t$  értékek.

## 3.2. Tárgyak mozgatása

Ebben a szakaszban egy konkrét alkalmazást nézünk meg a Delaunay-gráfok kapcsán, melyet Chew írt le cikkében részletesebben ([Ch86]). Az ilyen típusú gráfokkal (igazából bármelyik általánosított fajtájával is akár) könnyedén modellezhetünk például hálózatokat, melyekre igaz, hogy síkbarajzolhatóak, és a különböző pontok közti távolságot, azaz az élek súlyát egy egyenes síkbeli szakasz hosszával adjuk meg.  $w(A, B) := |AB|$ , ahol  $A, B \in \mathbb{R}^2$ .

Például van egy kiinduló  $s$  pontunk a síkban, és onnan szeretnénk valamilyen tárgyat elmozgatni egy  $d$  célpontba. Képzeljük ezt a tárgyat egy pontbeli dolognak. Azonban a síkon vannak akadályok, melyek most legyenek tetszőleges sokszögek, melyek nem metszik vagy fedik egymást, és csak véges sok van belőlük. Legyen ezek halmaza  $R$ . Célunk, hogy a tárgyunkat eljuttassuk  $s$ -ből  $d$ -be a lehető legrövidebb úton anélkül, hogy az akadályokkal ütközne.  $R$ -ből fogunk egy gráfot csinálni úgy, hogy a csúcshalmazunk álljon a sokszögek csúcsaiból, valamint  $s$  és  $d$ -ből, összesen  $n$  csúcsunk van. A sokszögek csúcsai pedig mindenképpen élei lesznek a gráfnak, mert azok mentén végigmehetünk, csak a sokszög belsejében nem, és optimálisan úgy kerülünk el egy akadályt, hogy a szélén mozgadjuk a tárgyunkat.

Első megközelítésként egy úgynevezett „láthatósági” gráfot mutatunk be, amelyben az előbb említett élek mellett azokat a csúcsokat kötjük össze, amelyek „látják egymást”, tehát amelyek között el lehet menni egyenes úton anélkül, hogy keresztülmennénk egy akadályon. Ez  $O(n^2)$  időben konstruálható, és Dijkstra-algoritmussal szintén  $O(n \log n)$  időben kereshetünk rajta legrövidebb utat ([We], [AAGHI]). Ezzel az optimális méretű utat fogjuk megkapni.

A második megközelítésben szerepelnek az  $L_1$ -Delaunay-gráfok. Vegyünk egy másik fajta háromszögelést, mely egy Delaunay-gráfhoz hasonlót fog adni, melyre a következők fognak teljesülni:

- (1,)  $G$  síkbarajzolható
- (2,) a  $G$ -beli minimális út hossza legfeljebb az optimális út hosszának  $\sqrt{10}$ -szerese, azaz spanner  $t = \sqrt{10}$  értékkel.

Lényegében egy olyan Delaunay-gráfot fogunk kapni, melybe  $R$  éleit mindenképpen bevesszük.

**3.2.1. Definíció.**  $T$  gráf álljon abból a korábban is említett  $n$  csúcsból a következő élekkel:  
-  $R$ -beli akadályok élei, és

- olyan  $u$  és  $v$  csúcsokat összekötő él, melyekhez tartozik egy körülírt  $C$  kör, melynek belsőjében, ha van más csúcs, akkor az nem „látható” sem  $u$ -ból, sem  $v$ -ből, tehát ez a másik belső csúcs el van választva  $u$ -tól és  $v$ -től egy  $R$ -beli éllel.

Ez a  $T$  gráf, melyet  $R$  függvényében hoztunk létre,  $O(n^2)$  időben konstruálhatjuk, de Chew megfogalmaz egy sejtést, miszerint mindez a legrosszabb esetben is csak  $O(n \log n)$  időt vesz igénybe. Ezek után a  $T$  gráfon Dijkstra-algoritmussal megtaláljuk a legrövidebb utat  $O(n \log n)$ -ben. Ez egy jó közelítés lesz a lehetséges legrövidebb úthoz ebben a problémakörben, hiszen Chew cikkében tételként is belátja, hogy ez egy  $\sqrt{10}$ -spanner.

Ezen problémakörnek konkrét gyakorlati alkalmazása például a robotikában van. Robotoknak „megtanítani”, hogy hogyan kerüljenek ki különböző akadályokat a térben egy az egyben modellezhető ezekkel a Delaunay-gráfok megközelítésekkel. Egy ezzel foglalkozó cikkben ([IIR]) oly módon készítenek Delaunay-gráfot és Voronoi-diagrammot, hogy magukat az akadályokat tekintik pontoknak, valamilyen kiterjedéssel. Ezen pontokhoz készítenek Voronoi-diagrammot, és utána keresnek ez alapján egy közelítő utat, amely során a robot nem fog ütközni egyik akadállyal sem.

# Irodalomjegyzék

- [AAGHI] Takao Asano, T. Asano, L. Guibas, J. Hershberger, H. Imai, Visibility-polygon search and Eukclidean shortest paths, *26. Annual Symposium on Foundations of Computer Science*, 1985, 155-164
- [ADDJS] I. Althöfer, G. Das, D. Dobkin, D. Joseph, J. Soares, On Sparse Spanners of Weighted Graphs, *Discrete and Computational Geometry* (9), 1993, 81-100
- [BCFMS] P. Bose, P. Carmi, M. Farshi, A. Maheswari, M. Smid, Computing the Greedy Spanner in Near-Quadratic Time, *Algorithmica*, 58, 2010, 711-729
- [BDLSV] P. Bose, L. Devroye, M. Löffler, J. Snoeyink, V. Verma, Almost all Delaunay triangulations have stretch factor greater than  $\pi/2$ , *Computational Geometry*, 44, 2, 2011, 121-127
- [Be] M. Bern, privát beszélgetés beszélgetés D. Dobkinnal, 1989
- [BGHP] N. Bonichon, C. Gavoille, N. Hanusse, L. Perkovic *The Stretch Factor of  $L_1$ - and  $L_\infty$ -Delaunay Triangulations*, 2012
- [BGM] P. Bose, J. Gudmundsson, P. Morin, Ordered theta graphs, *Computational Geometry* (28), 2004, 11-18
- [CD] L. P. Chew, R. L. Drysdale, Voronoi diagrams based on convex distance functions, *1. Symposium on Computational Geometry*, 1985, 235-244
- [Ch86] L. Paul Chew, There is a Planar Graph Almost as Good as the Complete Graph *Proceedings of the Second Symposium on Computational Geometry*, 1986, 169-177
- [Ch89] L. Paul Chew, There are planar graphs almost as good as the complete graph *Journal of Computer and System Science*, 1989, 205-219

- [ES] W. Evans, L. M. de Arruda Siaudzionis, On path-greedy geometric spanners, *Computational Geometry* 110, 2023
- [GGN] J. Gao, L.J. Guibas, A. Nguyen, Deformable spanners and applications, *Computational Geometry* (35), 2006, 2-19
- [Hw] F. K. Hwang, An  $O(n \log n)$  algorithm for rectilinear minimal spanning trees, *Journal of the ACM*, (26), 1979, 177-182
- [IIR] Ioan Ileana, Ioana Maria Ileana, Corina Rotar , *Robot motion planning using Delaunay triangulation*, 2004
- [KG] J. Mark Keil, Carl A. Gutwin *Classes of Graphs Which Approximate the Complete Euclidean Graph*, 1992
- [LS] D. T. Lee. B. Schachter, Two algorithms for constructing a Delaunay triangulations, *International Journal of Computer and Information Science*, 9:3, 1980, 219-242
- [MM] Joseph S.B. Mitchell, Wolfgang Mulzer, Proximity algorithms, *Handbook of Discrete and Computational Geometry*, 2017, 849-874
- [SH] M. I. Shamos, D. Hoey, Closest-point problems, 16. *IEEE Symposium on Foundations of Computer Science*, 1975, 151-162
- [We] E. Welzl, Constructing the visibility graph for n line segment in  $O(n^2)$  time, *Information Processing Letters*, 1985, 167-171
- [Xia] Ge Xia, Improved upper bound on the stretch factor of delaunay triangulations, 27. *Annual ACM Symposium on Computational Geometry (SoCG)*, 2011, 264-273
- [XZ] G. Xia, L. Zhang, Toward the tight Bound of the stretch factor of the Delaunay triangulations, 23. *Canadian Conference on Computational Geometry (CCCG)*, 2011