

EÖTVÖS LORÁND TUDOMÁNYEGYETEM  
TERMÉSZETTUDOMÁNYI KAR

# A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

SZAKDOLGOZAT  
FÖLDTUDOMÁNYI ALAPSZAK

*Készítette:*

Kerkovits Krisztián András  
térképész és geoinformatikus szakirányú hallgató

*Témavezető:*

Dr. Györffy János  
docens  
ELTE Térképtudományi és Geoinformatikai Tanszék



Budapest, 2012

# Tartalomjegyzék

	oldal
Előszó .....	3
1. fejezet: Alapfogalmak .....	4
2. fejezet: A felhasználható vetületek .....	6
2.1 A gnomonikus vetület .....	6
2.2 A szögtartó Cahill-vetület .....	7
2.3 Cahill kvázi-területtartó vetülete .....	9
2.4 A Cahill–Keyes-vetület .....	12
2.5 Snyder területtartó vetülete .....	15
2.6 Módosított Collignon-vetület .....	16
2.7 A Waterman-vetület .....	18
3. fejezet: Az oktaéder vetületeket ábrázoló programról .....	21
3.1 Bevezető .....	21
3.2 Jelölések .....	22
3.3 Általános számítási feladatok .....	22
3.4 Vetületi egyenletek .....	24
4. fejezet: A pillangó-vetületek mai lehetséges alkalmazásai .....	30
Összefoglalás .....	32
Köszönetnyilvánítás .....	33
Forrásjegyzék .....	34
Ábrák forrásai .....	35
1. melléklet: Az egyes leképezések összevetése .....	36
2. melléklet: A PHP-programok forráskódjai .....	37

## **Előszó**

Szakedolgozatom célja, hogy a kevésbé elterjedt pillangó típusú vetületeket, és az ebben készült térképeket bemutassa. Ezek a térképek esztétikai értékükkel, kedvező torzulási viszonyaikkal sokkal nagyobb népszerűségnek is örvendhetnének.

A szakdolgozat elkészítése során nehézséget okozott, hogy a témában kevés tudományos munka született, és azok is szinte mind angol nyelvűek voltak. Ezzel szemben hatalmas segítség volt, hogy Gene Keyes a honlapján összegyűjtötte a legtöbb ilyen tárgyú publikációt, így a kutatáshoz komoly alapot biztosított.

A munka elején leírom a téma szempontjából fontos geometriai és vetülettani fogalmakat. A dolgozat második részében ismertetem az egyes leképezéseket, előnyös és hátrányos tulajdonságaikat. A harmadik rész ezen leképezések vetületi koordinátáinak kiszámításához szükséges összefüggéseket mutatja be. Ezek alapján készült az a PHP-nyelvű program, mellyel a kiválasztott vetületű térkép könnyen megrajzoltatható. További két oldalt szenteltem mai lehetséges alkalmazások bemutatásának. Mellékletként megtalálható a mű végén egy összehasonlító táblázat és az előbb említett PHP-program forrása.

# 1. fejezet

## Alapfogalmak

### **Poliéderek és vetülettani vonatkozásaik**

Azokat a testeket, melyeket csak sokszöglapok határolnak, poliédereknek nevezzük, a határoló felületet pedig poliéder felületnek, vagy ezt is egyszerűen poliédernek hívjuk. A térképészet egyik célja, hogy a Földet síkban ábrázolja. A poliéderek ehhez nyújtanak segítséget. A poliéderek mint felületek lapjai egymás mellé síkban kiteríthetők, így ha egy térképvetület képfelületeként poliédert alkalmazunk, sík térképet készíthetünk. Egy jól megválasztott poliéder „közelebb áll” a Föld alakjához, mint a sík, kúp vagy henger, így kisebbek lehetnek a torzulások, mint más képfelület esetén. A poliéder vetületek sajátosságai a szakadások, amelyek azok mentén az élek mentén jelentkeznek, amelyek mentén a testet a kihajtogatás céljából felvágtuk. Általánosságban elmondhatjuk, hogy minél több lapja van egy poliédernek, annál „közelebb áll” a gömbhöz, tehát annál kisebbek lesznek a torzulások, viszont a felvágás miatt annál több lesz a szakadás. Poliéder vetületeket mind topokartográfiai (pl. III. katonai felmérés), mind geokartográfiai célokra felhasználhatunk.

### **Poliéder képfelülethez felhasználható leképezések**

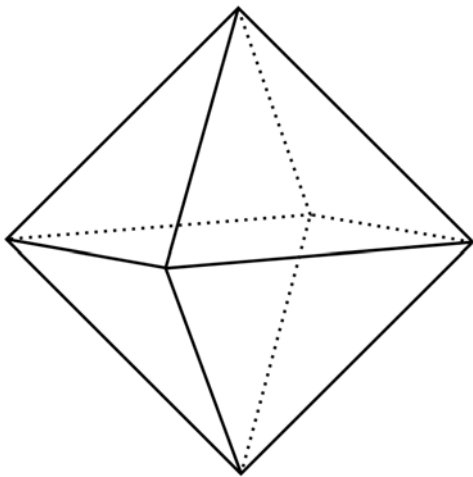
A poliéder vetületek speciális leképezéseket igényelnek. Először a földgömböt főkörök mentén felbontjuk a poliéder lapjainak megfelelően. Minden egyes gömbi sokszög a leképezést követően a poliéder egy-egy lapjába kell átmenjen; így a leképezéssel szemben az első követelményünk, hogy a határoló gömbi főkörívek képe egy szakasz (a poliéder egy éle) legyen. További követelményünk, hogy az érintkező sokszögek közös oldalai mentén ne szakadjon meg az ábrázolás: például ha egy fokhálózati vonal eléri a sokszög határát, a vonal a szomszédos sokszögön ugyanott folytatódjék.

### **Az oktaéder tulajdonságai vetülettani szempontból**

Az oktaéder egy olyan poliéder, melynek nyolc lapja van, és mind a nyolc lapja egybevágó, szabályos háromszög. (1. ábra) Az oktaéder bármely lapjának ösképe a gömbön egy olyan gömbháromszög, melynek minden szöge derékszög, és minden oldalához a gömb középpontjából nézve derékszög tartozik. Ezt az előnyös tulajdonságot még jobban kihasználhatjuk, ha az oktaéder két átellenes csúcsát a Föld forgástengelyére illesztjük.

## A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

Ezt poláris elhelyezésnek hívjuk. Ilyenkor a földgömböt az Egyenlítő, valamely szabadon megválasztott meridián, és ezen meridiántól 90 fokként három másik meridián mentén kell felosztanunk. A felosztás módját 11. ábrán a térkép keretén kívüli rajz segít megérteni. A gömbháromszög középpontját gömbháromszögtani összefüggésekkel meghatározhatjuk: poláris elhelyezés esetén a kiválasztott határoló meridiántól 45 fokra lévő hosszúsági körön, illetve a  $\pm\left[90^\circ - \arccos\left(\frac{1}{\sqrt{3}}\right)\right]$  szélességi körön található. Ezt a pontot érdemes segédpólusnak megválasztani.



1. ábra: Oktaéder



2. ábra: Poláris elhelyezésű oktaéderre leképezett Föld. Madarassy nyomán.

## A pillangó alakú vetületek

A térképészetnek nem csak az a fontos, hogy az előállított térkép pontos legyen, hanem hogy legyen valamiféle esztétikája. Az oktaédert Cahill úgy hajtogatta ki, hogy a teljes térkép egy pillangóra hasonlít. Példáját más emberek is követték. Az így nyert térkép szimmetrikus, a szakadások többnyire a déli féltekére esnek. Jól megválasztott középmeridiánnal elérhetjük, hogy ezek a szakadások az ábrázolás szempontjából kevésbé fontos területeket, például óceánokat vágjanak át. A következőkben olyan leképezéseket mutatok be, amelyek teljesítik a poliéder vetületekkel szemben korábban megfogalmazott elvárásokat, és így alkalmasak oktaéder képfelületre.

## 2. fejezet

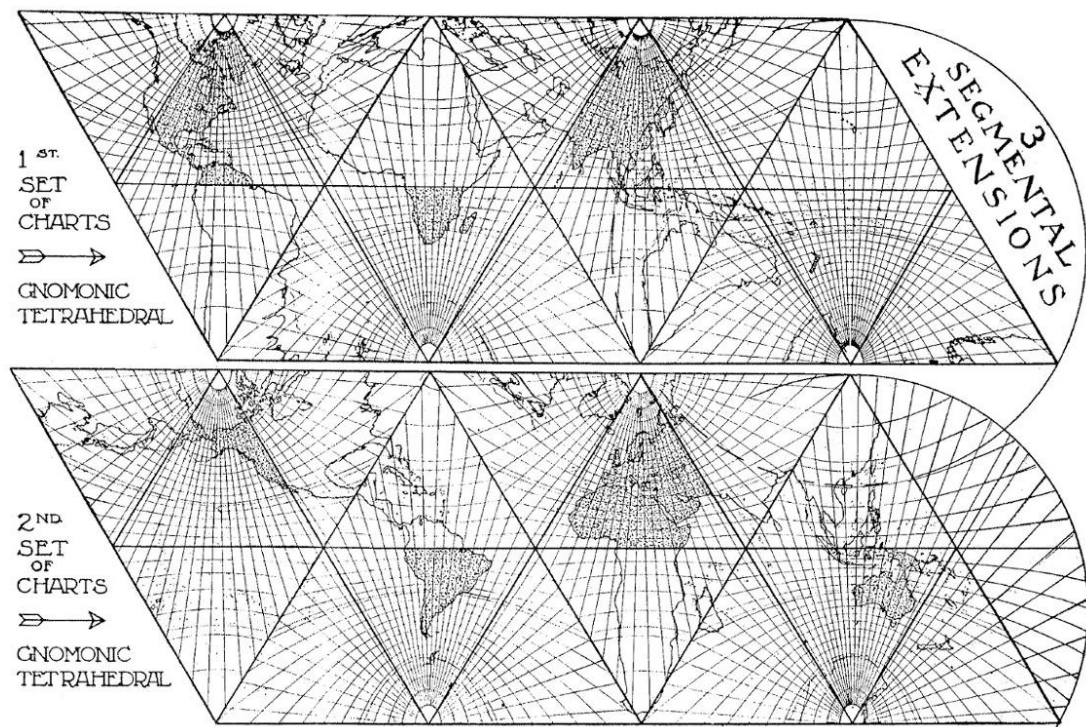
### A felhasználható vetületek

#### 2.1 A gnomonikus vetület

Valamennyi poliéder vetület legegyszerűbb megoldása a gnomonikus vetület, mely bármely gömbi főkört egyenesre képez le, ebből adódóan a gömb minden főkörök mentén történő felbontása esetén a részterületek képe sokszög.

#### Torzulási viszonyok

A gnomonikus vetület rendkívül előnytelen torzulási viszonyairól ismert. A vetület általános torzulású, és érintő helyzetben csak a segédpólusban torzulásmentes. A segédpólustól távolodva a torzulások rohamosan nőnek, oktaéder képfelület esetén a hossztorzulási modulus csúcspoknál eléri a 3-at. A paralellkörök képei kúpszeletek, az éleknél nagy törésekkel, de szakadás nélkül folytatódnak.



3. ábra: Gnomonikus oktaéder (szürke kontinensekkel) tetraéder vetületté (fehér kontinensekkel) kiterjesztve. Cahill nyomán.

A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

## **A vetület használata**

A gnomonikus vetület alkalmazását oktaéderre Cahill javasolja egy cikkében. Előnye az egyenesek megtartásából ered. Ezért elsősorban rádiózási és léginavigációs célokra ajánlja, ahol fontos a gömbi főkörök gyors megtalálása. Az éleken átmenő mérési feladatok esetén a fokhálózati vonalak meghosszabbításával tetraéder vetületté alakítható. (3. ábra)

## **A poláris helyzetű oktaéder előnyei**

Cahill többi vetületéhez hasonlóan poláris helyzetű oktaéderen rajzolta meg a gnomonikus vetületet, melynek középmeridiánja a Ny. h.  $22^{\circ}30'$ . A poláris helyzetű oktaéder olyan poliéder, amelyen a gnomonikus vetület fokhálózati képe valamennyi lapon egybevágó. A kockánál és a tetraédernél kedvezőbbek a torzulások. A nyolc hossztartó pont az É. sz. ill. D. sz.  $35^{\circ}15'51,8''$  és a K. h.  $22^{\circ}30'$  metszéspontjaiban, valamint ettől 90 fokként keleti és nyugati irányban találhatók.

## **2.2 A szögtartó Cahill-vetület**

Cahill szögtartó vetületét 1929-ben ajánlotta a meteorológusok számára.

### **A probléma**

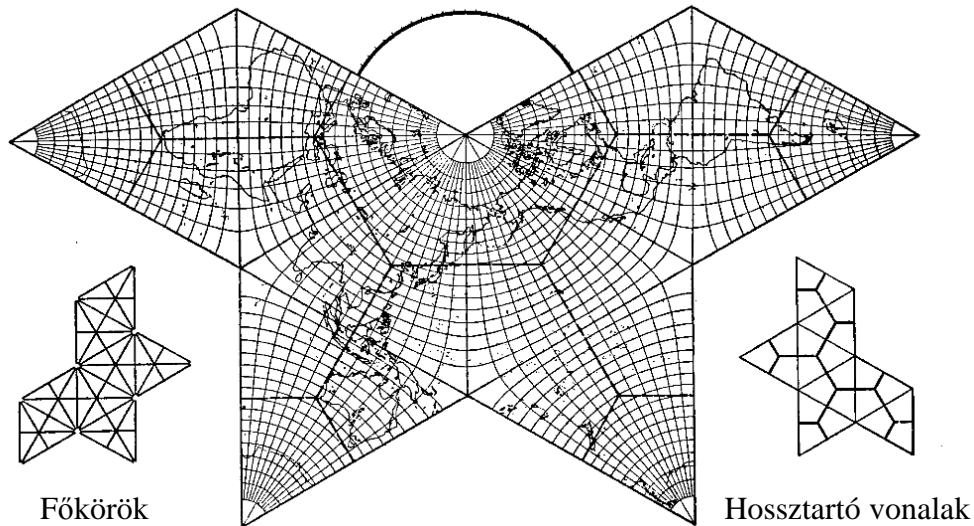
A meteorológusoknak olyan szögtartó vetületre volt szükségük, amely az egész Földet kis torzulásokkal képezi le. A legismertebb szögtartó vetületek egyike sem alkalmas erre. Míg a sztereografikus síkvetület csak a sarkokat, a Gauss–Lambert-kúpvetület csak a közepes szélességeket, addig a Mercator-hengervetület csak az Egyenlítő környékét ábrázolja pontosan. Ezen felül további problémát okoz, hogy a sík- és kúpvetületek gyakorlatilag csak a félgömb leképezésére alkalmasak, az Egyenlítőt körívre képezik le: a két féltékéről alkotott kép csak egy ponton illeszthető össze.

### **A szögtartó Cahill-vetület tulajdonságai**

A Cahill által javasolt vetület pillangó alakú, képfelülete egy oktaéder. A vetületi egyenletet Dr. Oscar S. Adams határozta meg. A vetület középmeridiánja Ny. h.  $22^{\circ}30'$ . Ezen oktaéder hat csúcspontján a szögtartás bár sérül, a térkép többi pontján szögeket torzulásmentesen képezi le. Így a meridiánok és a parallelkörök képei derékszögekben metszik egymást. A meridiánok képei megközelítően egyenesek, a parallelkörök képei pedig deriválható görbe vonalak. Kivétel az Egyenlítő, amelynek a képe a csatlakoztatás érdekében

## A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

egy töröttvonal. Továbbá nyolc meridián képe szintén kiegyenesedik: a középmeridián, és innen számolva 45 fokként minden meridián. A torzulások a háromszögek csúcsainál a legnagyobbak, a háromszög középpontját az oldalak felezőpontjával összekötő szakaszok hossztartó vonalak.



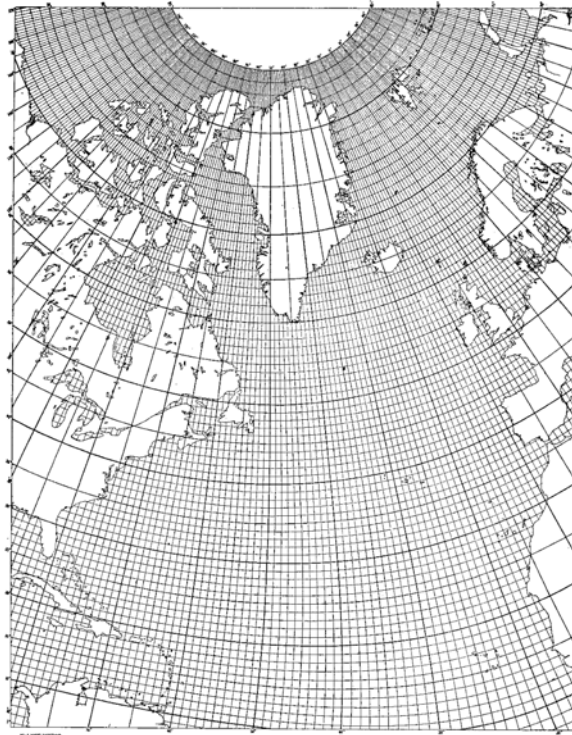
4. ábra: Szögtartó pillangó-vetület, a vastag vonalak hossztartóak. Cahill nyomán.

### A vetület előnyei

A vetület szögtartása miatt kifejezetten alkalmas meteorológiai számítások végzésére. Cahill a térképén fel akarta tüntetni a foktrapézok területét, az irányok pedig közvetlenül leolvashatóak. Az oktaéder csúcsainak környezete, ahol egyrészt a legkedvezőtlenebbek a torzulások, másrészt a vetületi koordináták kiszámítása problémás, mind a sarkokra, vagy az óceánok közepére esnek. Ezeken a helyeken a vetület létrehozásának korában nem folyt rendszeres meteorológiai mérés, így ez egyáltalán nem zavaró. A térkép nagyobb méretarányokban is használható regionális számításokhoz, mert a torzulások mindenütt kicsik. Ez lehetővé teszi, hogy a nemzeti meteorológiai szolgálatok ebben a vetületben tüntessék fel méréseiket. A nemzetközi szervezetnek a közös vetület következtében csak annyi dolga van, hogy a méretarányt lekicsinyítse, és az adatokat generalizálja. Szintén a kis hossztorzulásoknak köszönhető, hogy a hőmérsékleti- és nyomásgradiensek a térképről nagy pontossággal leolvashatóak.



A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben



5. ábra: A szög tartó vetület nagyobb méretarányban is szép képet ad. Cahill nyomán.

### **A pillangó alak kihasználása**

A pillangó-térképek jellegzetessége, hogy a sarkokon a meridiánsereg csak  $240^\circ$ -ot fed le. Cahill azt javasolta, hogy a maradék  $120^\circ$ -on a Föld felét ismételten képezzék le, és ezen az előző napi méréseket ábrázolják. Ez segít a változások követésében.

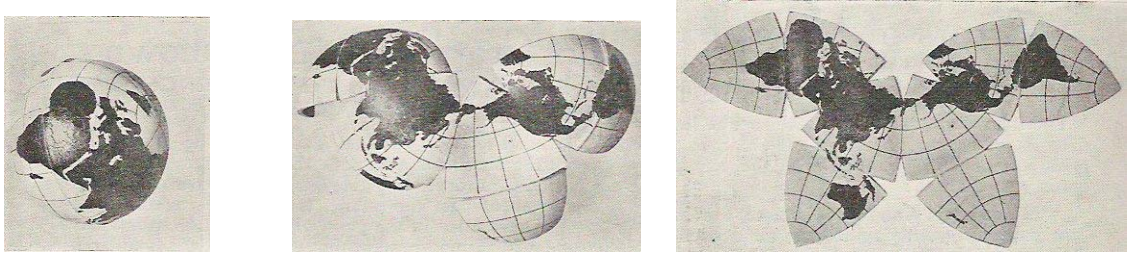
### **2.3 Cahill kvázi-terület tartó vetülete**

1913. február 25-én védte le az USA szabadalmi hivatala Cahill pillangó-vetületének ezt a változatát.

#### **A vetület szemléletes levezetése**

Cahill a vetületét érdekes módon mutatta be: Egy gumilabdára rárajzolta a kontinenseket, majd felvágta a megfelelő gömbi főkörök mentén. Az így felvágott gumilabdát üveglap alá téve ki lehetett lapítani. Így látható, hogy a vetület képfelülete annyira kevésbé tér el a gömbtől, hogy a gumi nyúlása ezt kompenzálni tudja. Cahill számos térképén kereten kívül elhelyezett sematikus ábrákkal is bemutatta kísérletét. (10. ábra)

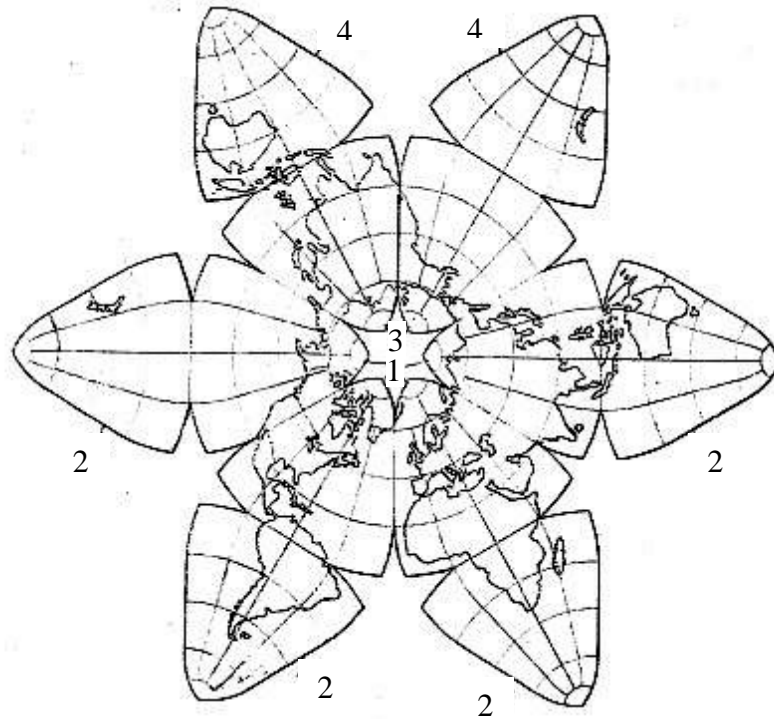
A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben



6., 7. és 8. ábrák: Cahill kísérlete a gumilabdával. Cahill nyomán

### Az oktaéder elhelyezése

Az oktaéder középmeridiánja a Cahillre jellemző Ny. h.  $22^{\circ} 30'$ . A határoló meridiánok képei görbe vonalak lennének, így a gömbháromszögek képei csak egy-egy ponton, az É. sz.  $41^{\circ} 48' 21''$  parallelszög mentén érintkeznének.



9. ábra: Cahill pillangó vetülete a szabadalomból. Cahill nyomán.

### A szakadások csökkentése

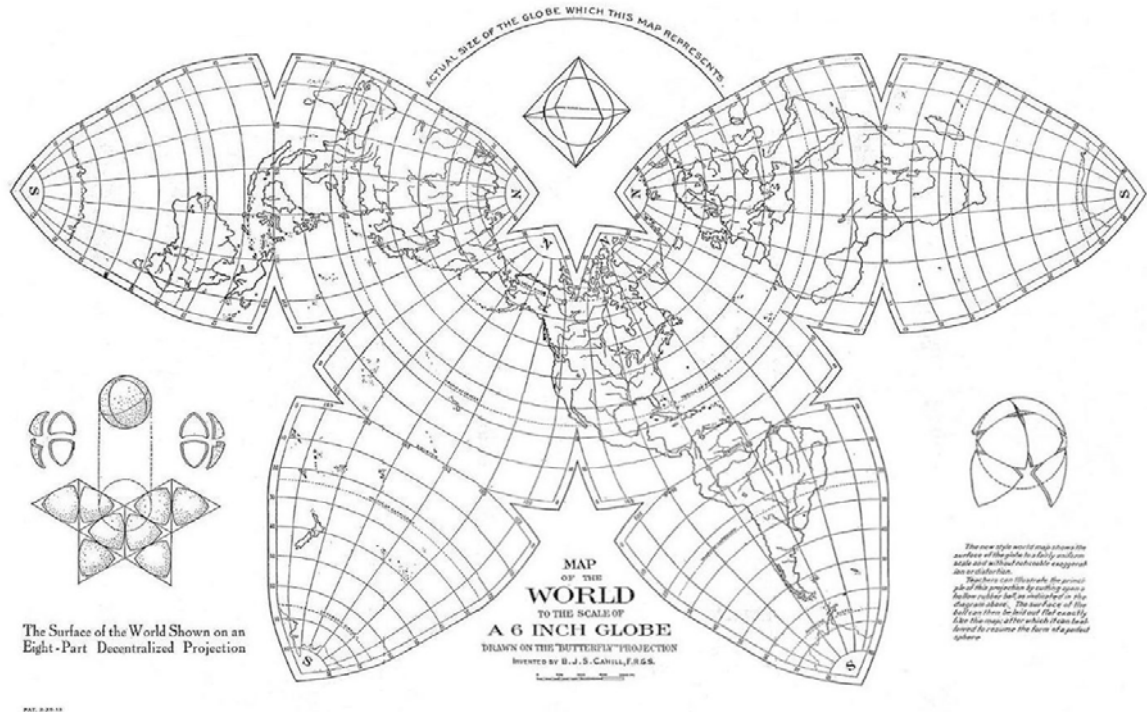
Mivel az így létrehozott vetület tele van szakadásokkal, ezeket torzításokkal kell csökkenteni. Cahill a szélső meridiánokat  $45^{\circ}$  hosszban — az É. sz.  $41^{\circ} 48' 21''$ -től északra és délre egyaránt  $22^{\circ} 30'$ -re — kiegyenesíti. Az Egyenlítőt szintén ki kellett egyenesíteni a háromszög közepén futó meridiántól  $22^{\circ} 30'$  távolságra mind keleti, mind nyugati irányban. Az így keletkezett alakzatokat az északi féltekén a 9. ábra 1-es számmal jelzi. Az ábrázolás így sem fedi le a teljes kört, tehát lehetőség van két-két háromszöget megismételni: ezeket a

A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

9. ábrán 3-as és 4-es számmal jelölve találjuk. Ennek köszönhető, hogy a Kamcsatka-félsziget nem szakad le Szibériától.

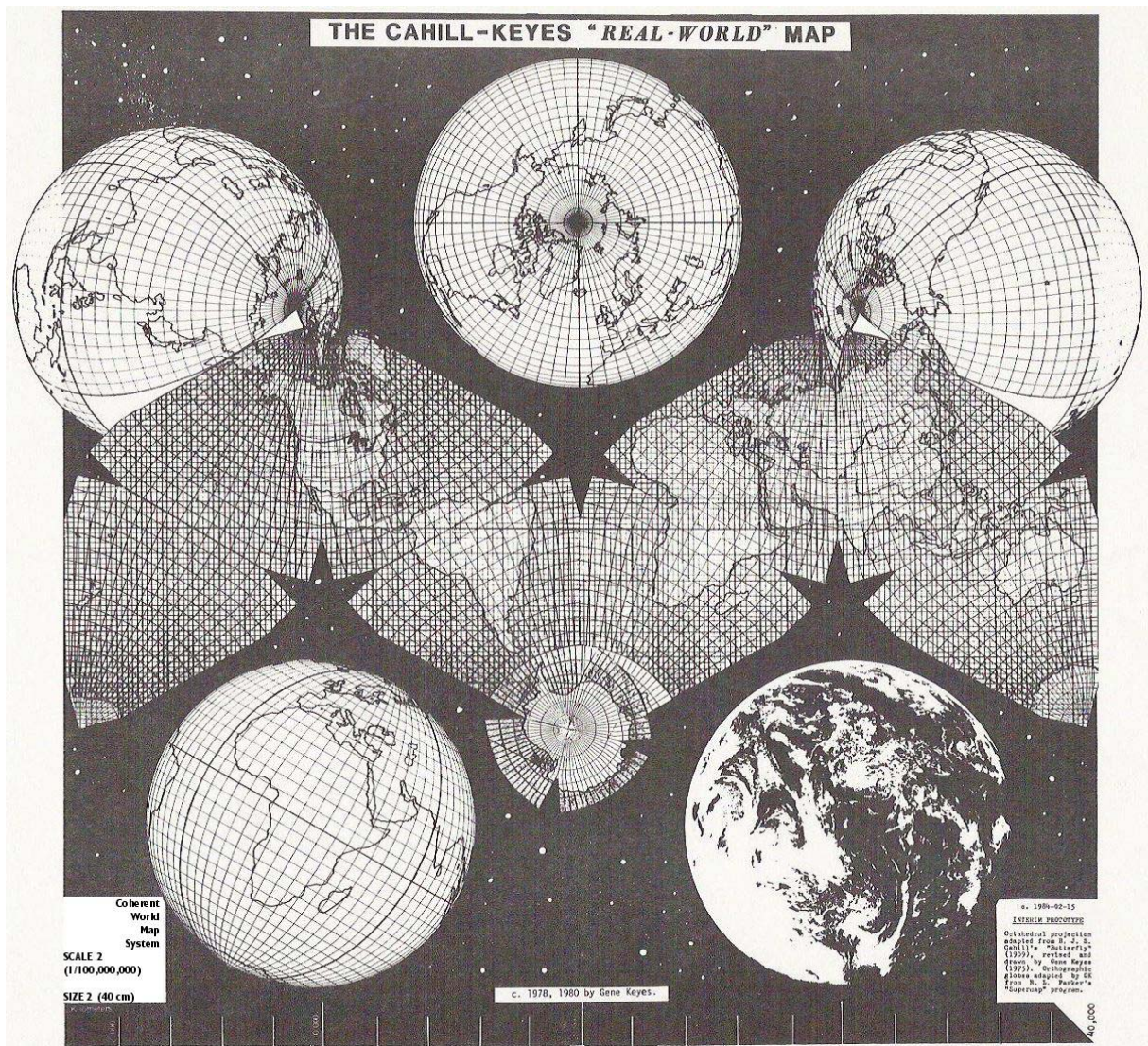
### További kiigazítások a szimmetria érdekében

A déli féltekén a határoló meridiánokat nem kellene kiegyenesíteni, hiszen itt a képek nem érintkeznek egymással. Cahill ennek ellenére úgy döntött, hogy a 9. ábrán 2-es és 4-es számmal jelölt alakzatok szélén futó meridiánokat a szimmetria kedvéért kiegyenesíti a D. sz.  $41^{\circ} 48' 21''$ -től mindkét irányban  $22^{\circ} 30'$ -re. Ez a megoldás azért praktikus, mert a térkép „háromszögei” az egybevágóság okán egymásra hajtogathatóak, és így a térképlapot könnyebb szállítani.



10. ábra: Cahill kvázi-területtartó térképe. A kereten kívüli magyarázó ábrák: fent a gömb és az oktaéder kapcsolata, jobb és bal oldalon lent a gumilabda kiterítése. Cahill nyomán

## 2.4 A Cahill–Keyes-vetület



11. ábra: Keyes világtérképe. Keyes nyomán

Keyes 1975-ben számos változtatást hajtott végre Cahill vetületén:

### A pillangó alak lecserélése

Abból a célból, hogy a megírások talpai dél felé nézzenek az egész térképen, Keyes M alakban illesztette össze az oktaéder lapjait. Így a térkép teteje a konvenciónak megfelelően észak felé néz, továbbá az Egyenlítő hosszát — melyet Keyes 40 000 km-nek vett — jobban szemlélteti.

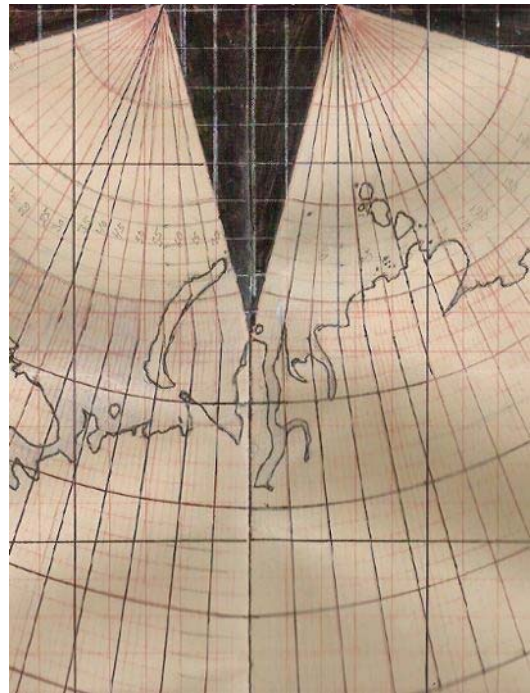
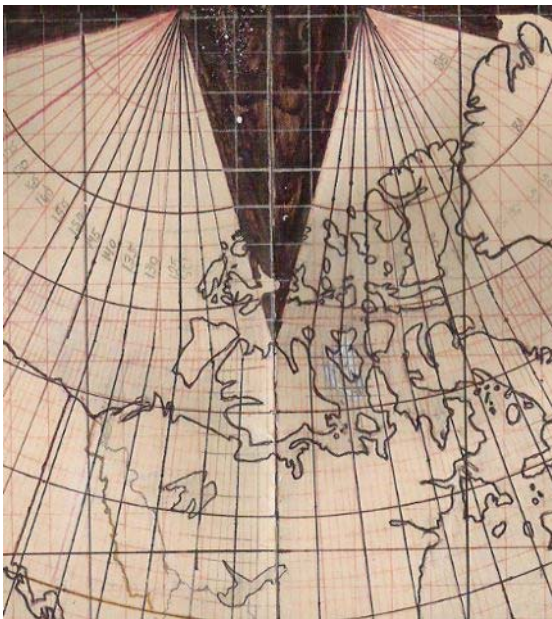
A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

### **Középmeridián eltolása**

Mivel a fokhálózatot nehéz kézzel kirajzolni a Cahill által választott Ny. h.  $22^{\circ}30'$ -es középmeridiánnal, és a lapok szélén lévő foktrapézok megszakadnak, Keyes a Ny. h.  $20^{\circ}$ -os meridiánt jelölte ki az oktaéder élének. Ez jobban megfelel az  $1^{\circ}$ -os és  $5^{\circ}$ -os fokhálózatnak. Cahillnél ez azért nem jelentett gondot, mert kisebb méretarányban dolgozott, ahol megfelelt a  $7^{\circ}30'$ -es fokhálózat is, amelynél a foktrapézok nem szakadnak.

### **A szakadások csökkentése a sarkoknál**

Cahill olyan vetületet használt, mely az élkre eső meridiánokat körívekre képezi le. A szakadások mérséklése érdekében ezeket kiegyenesítette  $45^{\circ}$  hosszban. Látszik tehát, hogy mind a sarkoknál, mind az Egyenlítőnél egy-egy  $22^{\circ}30'$ -es íven az ábrázolás megszakad. Hogy Szibériát és a Viktória-szigetet ne kelljen megszakítani, Keyes módosítja ezt az alakot úgy, hogy csak a  $17^{\circ}$ -os pólustávolságig érjen a bevágás.



*12. és 13. ábrák: Szakadások a Viktória-sziget és a Jamal-félsziget környékén. Keyes nyomán.  
Érdemes összehasonlítani Waterman vetületével (19. és 20. ábrák)*

A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

## **A fokhálózat átrajzolása**

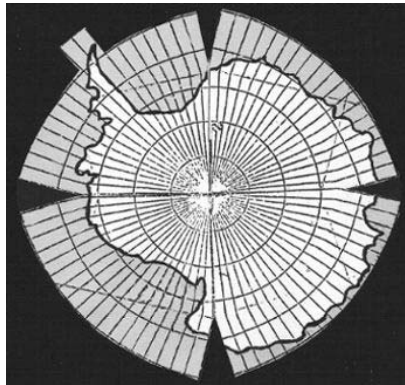
Keyes arra törekedett, hogy a fokhálózat képe egyenletes legyen abban az értelemben, hogy ne legyen hirtelen ugrásszerű különbség a paralelkörök képeinek távolságában. Ennél fogva a vetület csak alakjában hasonlít Cahillére, az egyes gömbnyolcadok leképezése jelentősen eltér.

## **Paralelkörök leképezése körökre a sarkoknál**

Keyes célja volt, hogy a sarki területek összeilleszthetőek legyenek. Esztétikai okokból a sarki régiókban a paralelköröket koncentrikus, egyenközű köröknek ábrázolja, amely gyakorlatilag megegyezik Postel síkvetületével.

## **Antarktisz egyesítése**

Mivel Cahill vetületében a meridiánok képei körívek, nem volt lehetőség az Antarktisz összefüggő ábrázolására az oktaéder lapjainak más elrendezésével sem. Ezt még gnomonikus vetületben sem lehetne megtenni, hiszen ekkor a sarkoknál a meridiánsereg nem  $360^\circ$ -ban, hanem csak  $240^\circ$ -ban töltene ki a síkot. A Cahill–Keyes-vetületben ezt meg lehet tenni, mert a sarkoknál Postel-síkvetületet használ. Keyes így az Antarktisz a térkép alján, az M alak közepén összefüggően tudja ábrázolni. A fokhálózatot a D. sz.  $65^\circ$ -ig minden irányban kirajzolja.



*14. ábra: Az Antarktisz Keyes térképén. Keyes nyomán.*

## **Szakadások feloldása**

Keyes a térképen keletkező szakadásokat gyakran bizonyos területek többszöri megrajzolásával oldja fel. Ilyen területek a Kamcsatka-félsziget, Izland, Grönland keleti partjai, az Erzsébet királynő szigetek, a Kerguelen-szigetek, a Salamon-szigetek keleti része és a Maldív-szigetek déli része.

A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

## ***2.5 Snyder területtartó vetülete***

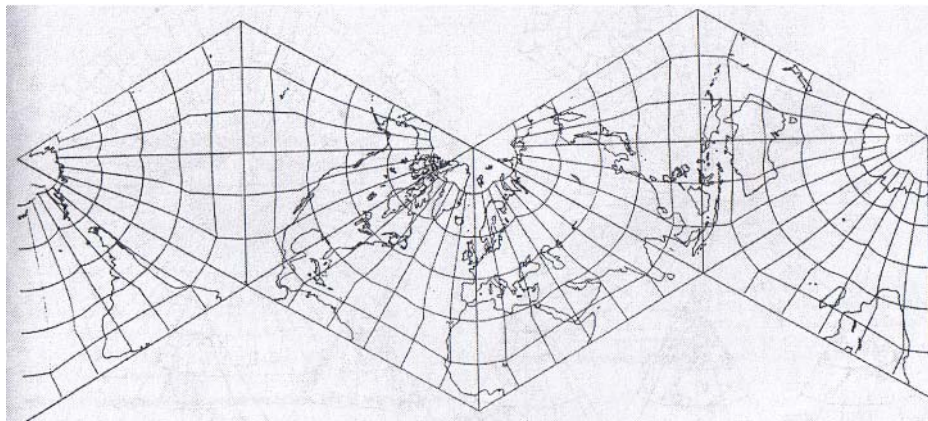
Snyder 1992-ben publikálta területtartó vetületét, mely alkalmas poliéder képfelületekhez.

### **A vetület levezetése**

Snyder a Lambert-féle síkvetületből indult ki. Ez ugyan területtartó, de nem alkalmas poliéderekre, hiszen a határoló gömbi főkörök képei nem egyenesek, a poliéder élein helyenként túlnyúlnak, vagy nem érik el azokat. Ha a sugárfüggvény kiigazításával ezt a problémát kiküszöböljük, a határoló vonalak kiegyenesednek, de a vetület már nem területtartó. A területtartást úgy nyerjük vissza, hogy a segédmeridiánok képeinek hajlásszögét is megfelelő mértékben módosítjuk.

### **Lehetséges képfelületek**

Snyder vetülete bármilyen poliéderre alkalmazható, melynek minden lapja szabályos sokszög. Az általa leírt képfelület a csonka ikozaéder, és ezen felül megadja a számítás módját valamennyi szabályos poliéderre, így az oktaéderre is.



*15. ábra: Snyder területtartó térképe oktaéderen. Snyder nyomán.*

### **Torzulások**

A vetületben a legnagyobb hossz- és szögtorzulások a segédpólusban találhatók. A segédpólust a csúcspontokkal összekötő szakaszok mentén a fokhálózati vonalak látványosan megtörnek, de az élek mentén nincs feltűnő törés. Az oktaéder képfelület hátránya, hogy fent említett törések jóval nagyobbak, mint csonka ikozaéder esetén. Ezzel szemben előnye, hogy a lapok középpontjai a földgömb középpontjától azonos távolságra vannak, így a méretarány valamennyi lapon megegyezik.

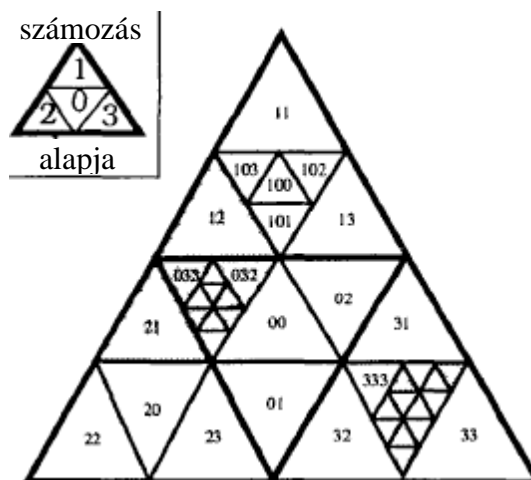
## 2.6 Módosított Collignon-vetület

### Származtatás

Keith C. Clarke 1995-ös publikációjában leírta, hogy a Collignon-féle képzetes hengervetület apró változásokkal alkalmas oktaéder képfelületre. Az oktaéderre alkalmas változatot a gyémánt alakú elrendezésből származtatjuk úgy, hogy középmeridiánjának megadjuk a leképezendő gömbnyolcad közepén átmenő meridiánt, majd a megfelelő háromszöget kivágjuk.

### A vetület célja

Lugo és Clarke a vetületet arra használta, hogy a térinformatikában megszokott négyzet alakú négy-fa modell helyett háromszögeken alapuló négy-fát tudjon leprogramozni. Ez a háromszög alapú szabályos tesszelláció 3D alkalmazásokhoz kitűnő, mert a TIN-hez hasonlóan háromszögekhez rendelve lehet domborzatmodellt készíteni. A vetület jellemzői miatt az azonos szinten lévő háromszögek azonos területet fednek le a földfelszínen, így egyenletes lesz az adatsűrűség az egész bolygón.



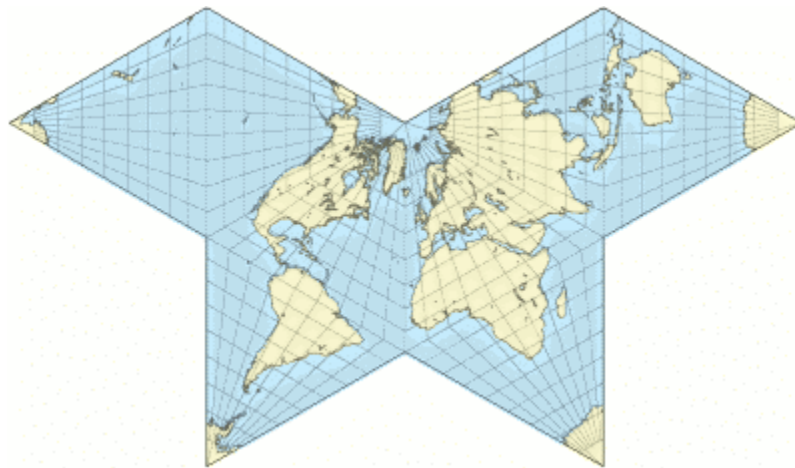
16. ábra: Háromszög alapú szabályos tesszelláció indexei. Lugo nyomán.



A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

### **Torzulások, törések**

Noha a Collignon-vetület területtartó, a Clarke által megadott módosított vetületi egyenletek már nem. Kiszámolhatjuk, hogy a vetület területtorzulási modulusa konstans  $\frac{\sqrt{3}}{2}$ . A területtartás érdekében Fehér Dániel mindkét koordinátát megszorozta  $\sqrt{\frac{2}{\sqrt{3}}}$ -mal. A Collignon-vetület jellegzetessége a nagy szögtorzulás, mely ebben az elrendezésben is feltűnik. A háromszögek éleinél a paralellkörök  $60^\circ$ -os szögben megtörnek, a meridiánok pedig az Egyenlítőnél szenvednek nagyobb töréseket. Csak a háromszög közepén futó meridián nem törik meg. Területtartó esetben  $\pm \arcsin\left(\frac{8}{\pi\sqrt{3}} - 1\right)$  szélesség mentén hossztartó.



17. ábra: Módosított Collignon-vetület. Furuti nyomán.

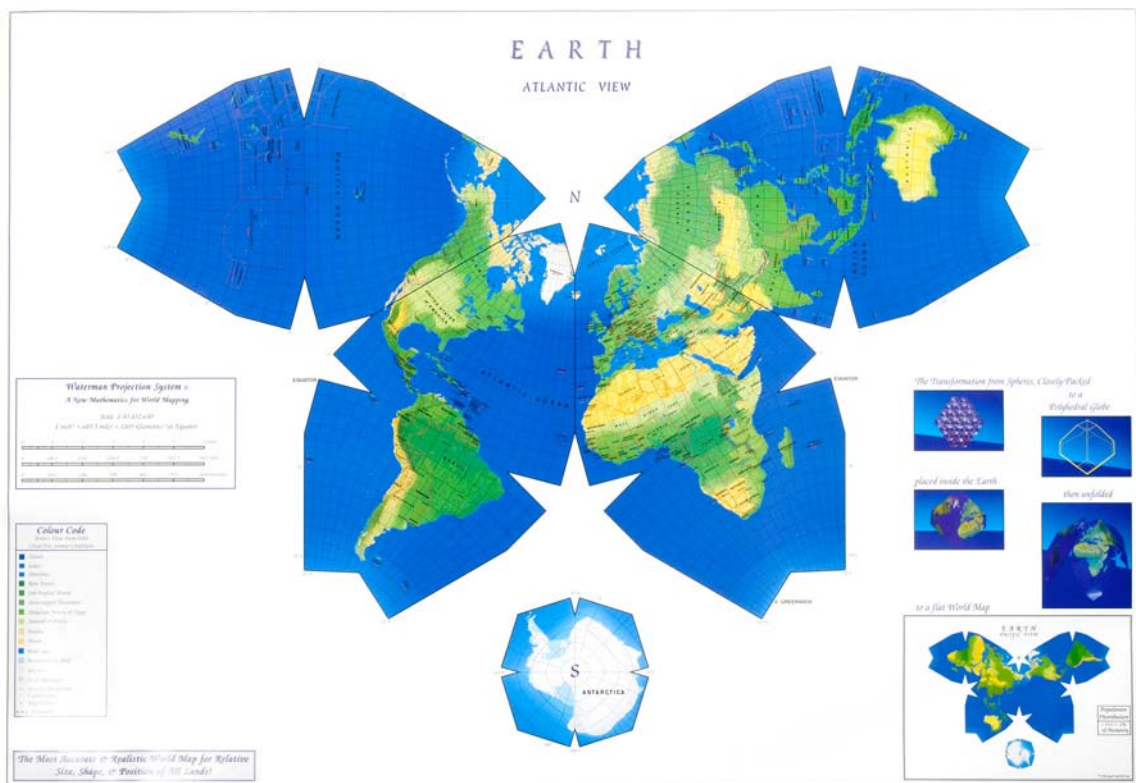
A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

## 2.7 A Waterman-vetület

Waterman 1996-ban készítette el az egész Földet ábrázoló színes falitérképét egy Cahilléhez hasonló vetületben.

### A vetület eredete

Waterman matematikus, és többek között azzal foglalkozik, hogy milyen poliéderben lehet gömböket a lehető leghelytakarékosabban elhelyezni. Az általa kiszámított speciális poliédereket Waterman-poliédereknek nevezzük. Waterman ezek közül kiválasztott egy nem-szabályos csonka oktaédert, és azt jelölte meg a vetület képfelületének. A Földet erre a nem-szabályos csonka oktaéderre területtartó módon képezi le.



18. ábra: Falitérkép Waterman-vetületben. Kereten kívül, a jobb alsó sarokban látható magyarázó ábrák rendre: egymás mellé helyezett gömbök; ebből levezetett poliéder; a poliéderre vetített Föld; a kihajtogatás módja; végül a kész térkép. Waterman nyomán.

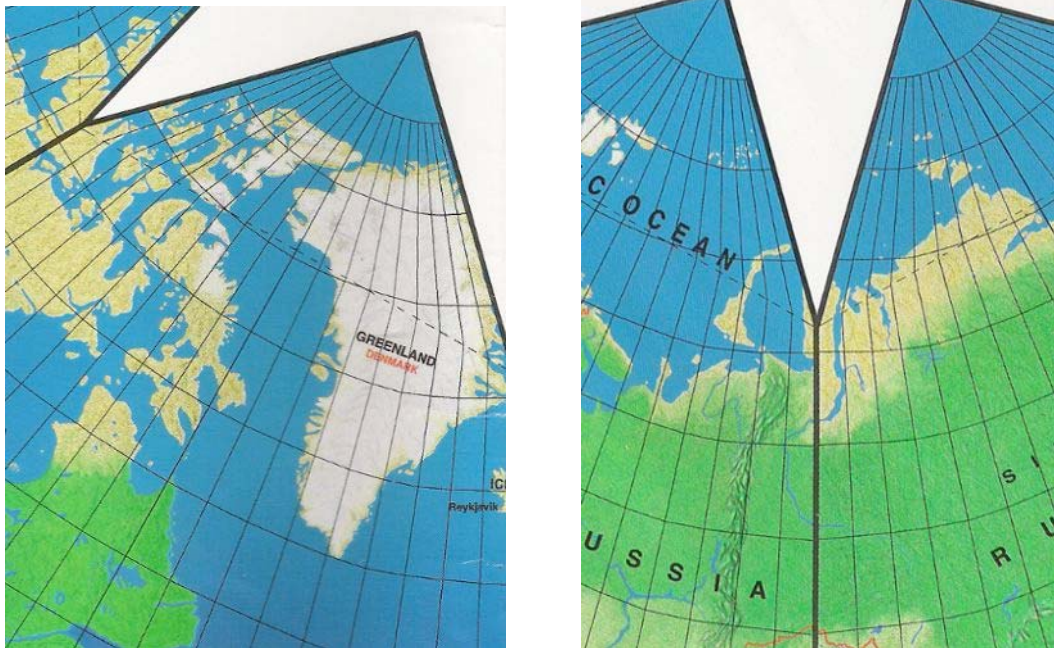
A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

## A pillangó alak

Noha Waterman a vetületét a Cahill–Keyes-vetülettől függetlenül alkotta meg, az eredmény hasonló megjelenésű. A középmeridiánt ő is a Ny. h. 20°-ban határozta meg. A képfelület négyzeteit az átlók mentén feldarabolta, és a pillangó alakban kiterített hatszögekhez kapcsolta.

## Szakadások

Az így megalkotott vetületben a Csukcsföld területén Eurázsia megszakad. Ez a probléma természetesen Keyes vetületén is jelentkezett, ám ő ezt a Kamcsatka-félsziget kétszeres kirajzolásával feloldotta. Waterman ezzel szemben a Föld minden részét csak egyszer ábrázolja. Ráadásul a felhasznált Waterman-poliéder tulajdonságai miatt a pólusoknál kb. 18°-os bevágások keletkeznek, amelyek olyan területeket is megszakítanak, amelyek Keyesnél összefüggők voltak. Ilyen például a Viktória-sziget.



19. és 20. ábrák: Szakadások a Viktória-sziget és a Jamal-félsziget környékén. Keyes nyomán.

*Érdemes összehasonlítani Keyes vetületével (12. és 13. ábrák)*

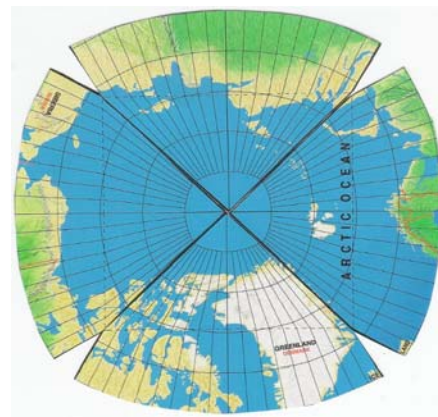
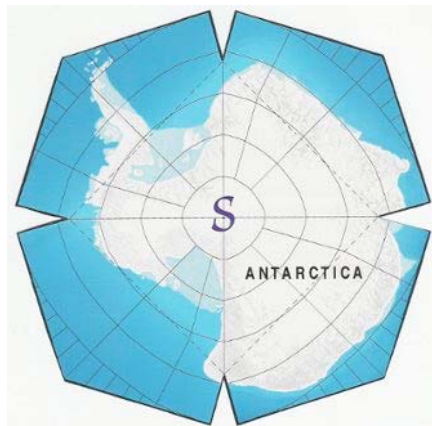
## Az Antarktisz összefüggő ábrázolása

Waterman az Antarktisz a térkép alján, minden más térképrésztől elkülönítve ábrázolja. Mivel itt is betartja azt a szabályt, hogy egy területet csak egyszer rajzol meg, a kivágott részek helyén a hatszögeket megcsonkítja.

A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

### **A fokhálózat képének problémái a sarkoknál**

A déli pólus környékén szembetűnő, hogy a négyzet átlói mentén a paralellkörök képei megtörnek. Ez a többi négyzetnél is így van, de ezeken nem látványos, mert az ábrázolás amúgy is megszakad; csak akkor láthatók a törések, ha a négyzeteket újból összeillesztjük. A meridiánok a négyzetek szélén törnek meg, jelentős mértékben torzítva Grönland alakját. Ezek a meridiánok a területtartás okán nem is egyenközűek, így számos Kanada északi partjainál fekvő sziget alakja is erősen torzul, holott teljes mértékben a sarki négyzet területére esnek. A torzulások a  $75^\circ$ -os szélességnél a legszembetűnőbbek. A meridiánok képei az Egyenlítőhöz közeledve már egyre kiegyenlítettébb rajzolatot mutatnak.



21. és 22. ábrák: Poláris területek Waterman-vetületben. Keyes nyomán.

## 3. fejezet

# Az oktaéder vetületeket ábrázoló programról

### 3.1 Bevezető

#### A program célja

Az oktaéder vetületek megjelenítéséhez programot készítettem PHP nyelven. A program feladata, hogy az egyes leképezések tulajdonságait a fokhálózat képe, a kontinensek alakja, mérete és helyzete segítségével bemutassa. Lehetőséget nyújt, hogy a középmeridián és a lapok kihajtási módjának megválasztása után lássuk, mely területek szempontjából optimális az ábrázolás, és mely területekre esnek a szakadások.

Két kivétellel bármely említett vetületet kipróbálhatjuk. A kvázi-területtartó Cahill-vetületben számos „kézi” torzítás is van, így nem írható le egzakt matematikai alakban. A Wateman-vetület egyenleteit nem sikerült megszerezniem.

#### Technikai háttér

A megjelenítés a GD könyvtár segítségével PNG formátumban történik. A program először felrajzolja a gömbnyolcadoknak megfelelő háromszögeket, erre rárajzolja a kontinensek töréspontjait, majd legvégül a fokhálózati vonalakat, melyek töréspontjait fokenként számolja ki.

## Oktaéder vetületek

<b>Vetület</b>	Cahill-Keyes ▾
<b>Alak</b>	<input checked="" type="radio"/> Pillangó <input type="radio"/> M alakú
<b>Középmeridián</b> <i>(Valós szám, -180 és 180 között.)</i>	-20 °
<b>Fokhálózat sűrűsége</b> <i>(Csak pozitív valós szám lehet.)</i>	10 °
<b>Méretarány</b> <i>(Csak pozitív valós szám lehet.)</i>	1 képpont = 50 km.
	<input type="button" value="Rajz!"/>

23. ábra: A program kezelőfelülete

A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

### 3.2 Jelölések

A következő jelöléseket alkalmazom ebben a fejezetben:

$\varphi, \lambda$ : A leképezendő pont földrajzi koordinátái.

$\varphi_0, \lambda_0$ : A segédpólus földrajzi koordinátái.

$\varphi^*, \lambda^*$ : Segéd földrajzi koordináták

$\rho, \omega$ : A pont vetületi koordinátái polárkoordináta-rendszerben.

M: Méretarány szám (méretarány reciproka)

R: A gömbnek képzelt Föld sugara (6371 km)

x, y: A pont vetületi koordinátái derékszögű koordináta-rendszerben.

$\alpha$ : Síkbeli forgatás szöge

$x_0, y_0$ : Az eltolás-vektor koordinátái

$x^*, y^*$ : A pont koordinátái a számítógép derékszögű koordináta-rendszerében.

### 3.3 Általános számítási feladatok

#### Segéd földrajzi koordináták számítása

Valamennyi töréspont vetületi koordinátáinak kiszámítása a következőképp történik: Először, ha szükséges, kiszámolom a pont segéd földrajzi koordinátáit az ismert trigonometrikus összefüggésekkel:

$$\varphi^* = \arcsin[\sin \varphi_0 \cdot \sin \varphi + \cos \varphi_0 \cdot \cos \varphi \cdot \cos(\lambda - \lambda_0)]$$

$$\lambda^* = \arccos\left(\frac{\sin \varphi - \sin \varphi_0 \sin \varphi^*}{\cos \varphi_0 \cos \varphi}\right),$$

ám ez csak a szög nagyságát adja meg. Előjele  $\sin(\lambda - \lambda_0)$  előjelével egyezik meg.

Poláris helyzetű oktaéder képfelület esetén a gömbnyolcad középpontját célszerű a fenti számítás segédpólusaként kijelölni:

$$\varphi_0 = \pm \left[ 90^\circ - \arccos\left(\frac{1}{\sqrt{3}}\right) \right],$$

az előjel a választott féltékétől függ.

$$\lambda_0 = m + 45^\circ,$$

ahol m a gömbnyolcad nyugati határán húzódó meridián szögértéke.

A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

### Derékszögű koordináták számítása

A vetületi koordináták kiszámítása az egyes vetületekben eltér, így ezt a következő alfejezetben ismertetem. Előfordul, hogy a vetületi egyenlet polárkoordinátákban adja meg a végeredményt. Mivel a PHP nem kezel polárkoordinátákat, ezeket a következőképp számítom át:

$$x = -\rho \cdot \sin \omega$$

$$y = -\rho \cdot \cos \omega$$

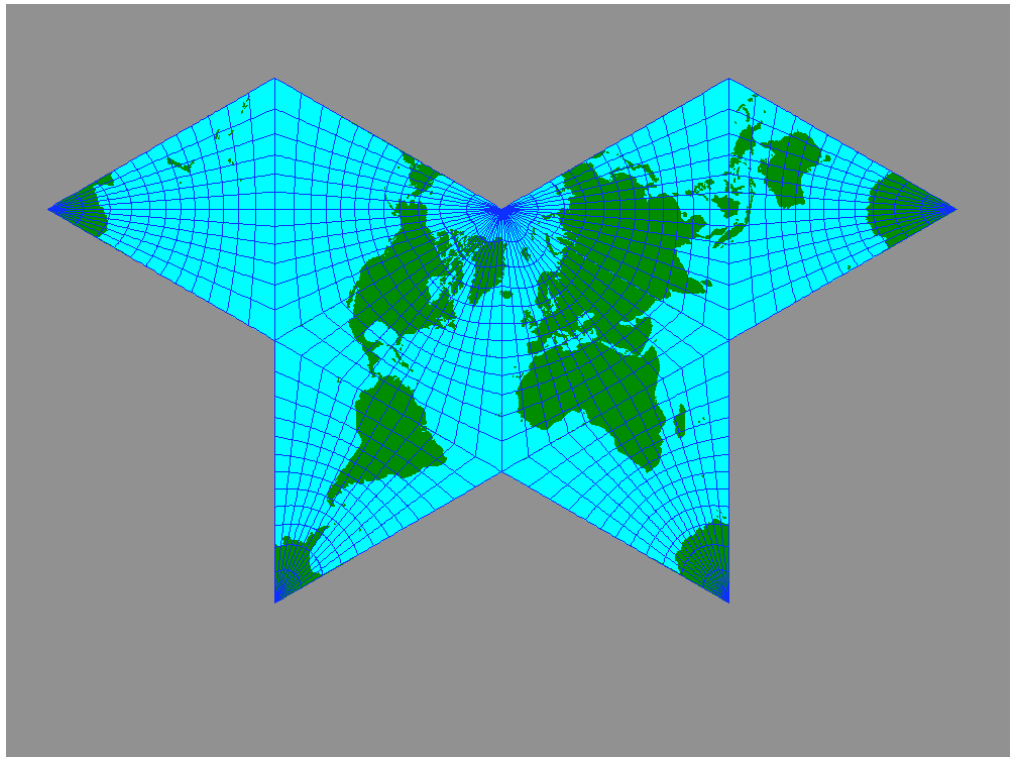
A geodéziában jártasabb emberek felismerik, hogy ez nem egyezik meg az általánosan használt átszámítási képlettel. Ennek oka, hogy a PHP koordináta-rendszere 90°-kal az óra járásával egyező irányban el van forgatva a geodéziában használt koordináta-rendszerhez képest. Tehát az x tengely kelet felé, az y tengely dél felé mutat.

### Síkbeli transzformációk

A koordinátákat az egyes háromszögek illesztése érdekében általában el kell forgatni. Végül hogy a háromszögek egymás mellé kerüljenek, el kell tolni a pontot. Ezt a két műveletet egyszerre is elvégezhetem:

$$x^* = x \cdot \cos \alpha + y \cdot \sin \alpha + x_0$$

$$y^* = y \cdot \cos \alpha - x \cdot \sin \alpha + y_0$$



24. ábra: Gnomonikus vetület. Középmeridián: Ny. h. 22° 30'; fókuszát 7° 30'

A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

### 3.4 Vetületi egyenletek

#### Gnomonikus vetület

A gnomonikus vetület egyenlete rendkívül egyszerű, ha a vetület középpontja a pólusban van. Ezért a számításhoz az előző alfejezetben kiszámolt segédföldrajzi koordinátákat használom:

$$\rho = \frac{R \cdot \operatorname{tg}(90^\circ - \varphi^*)}{M}$$

$$\omega = \lambda^*$$

#### Szögtartó vetület számítása Lee komplex függvényei segítségével

A szögtartó vetületek közül a legegyszerűbb a sztereografikus síkvetület. A feladat megoldásához a ferde tengelyű, a *segédegyenlítő síkjában* elhelyezett képsíkú változatot kell használni. A sztereografikus koordinátákat egységsugarú gömbön számítom.

$$\rho = \operatorname{tg} \frac{90^\circ - \varphi^*}{2}$$

$$\omega = \lambda^*$$

Ezután derékszögű koordinátákat számítok, melyeket  $p$  és  $q$  betűvel jelölök.

$$p = \rho \cdot \cos \omega$$

$$q = \rho \cdot \sin \omega$$

A sztereografikus vetület, bár szögtartó, önmagában nem használható poliéder képfelületekhez, mert a gömbi főköröket körívre képezi le. Ezért van szükség olyan függvényekre, melyek szögtartó módon szabályos háromszögekbe képezik bele ezeket az alakzatokat. Lee ehhez vezette le a következő megoldást.

A sztereografikus koordinátákat a további számításokhoz a  $z$  komplex számmal jellemzem:

$$z = p + i \cdot q$$

Mivel a PHP nem kezel komplex számokat, így ezeket kételemű tömbként tárolom. A szükséges műveleteket (összeadás, szorzás, hatványozás) az alapvető összefüggések segítségével elkészített saját függvényekkel oldom meg.

A koordinátákat a Schwarz-integrál felhasználásával kiszámolt komplex polinommal közelítjük:

$$r = 2\sqrt{2}z$$

$$w = r + 0,026042r^4 + 0,003193r^7 + 0,000555r^{10} + 0,000113r^{13} + 0,000025r^{16} + 0,000006r^{19} + 0,000002r^{22}$$



A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

A  $w$  komplex szám megfelelő részei lesznek az új koordináták. A PHP koordináta-rendszere miatt a képzetes rész ellentettje lesz az  $x$ , míg a valós rész ellentettje az  $y$  koordináta. A koordinátákat a méretaránynak megfelelő konstanssal kell szorozni. Gyök kettővel azért osztok, hogy a térkép összevethető legyen a többivel. (Az érintő gnomonikus vetülethez hasonlóan így a segédpólusban 1 lesz a hossztorzulás.)

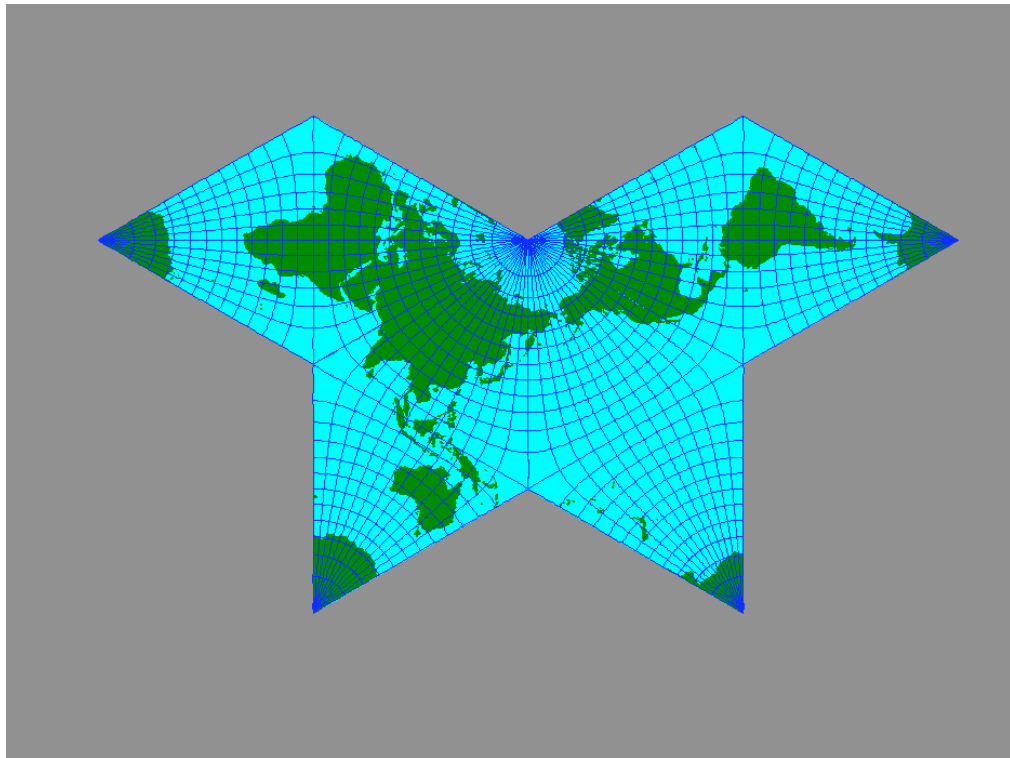
$$x = -\frac{R}{\sqrt{2M}} \Im w$$

$$y = -\frac{R}{\sqrt{2M}} \Re w$$

Ezeket a koordinátákat a lassú konvergencia miatt a segédpólustól távol eső területeken már nem használhatom, mert feltűnően nagy hibákhoz vezet. Ezért a háromszögek csúcsainál (azok  $2^\circ 30'$ -es környezetében) a koordinátákat egy lineáris képlettel közelítem.

Az együtthatókat igyekeztem a következő céloknak megfelelően megválasztani: Legfontosabb, hogy a gömbháromszög csúcsainak képei egy szabályos háromszög csúcsait adják ki. Az Egyenlítő képe egyenes kell, hogy maradjon. Semelyik fokhálózati vonal nem törhet meg. Végül, hogy az Egyenlítő a meridiánokkal lehetőleg derékszöget zárjon be.

Mivel ezeket a célokat ránézésre próbáltam összehangolni, az eredmény vetületanalízis szempontjából ezeken a területeken használhatatlan, viszont szemléltetésre jól felhasználható.



25. ábra: Szögtartó vetület. Középméridián:  $K. h. 157^\circ 30'$ ; fokhálózat  $7^\circ 30'$

A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

### Snyder területtartó vetülete

Snyder területtartó vetületéhez meg kell határozni, hogy mekkora az az oktaéder, amelynek a felszíne megegyezik a Föld felszínével. Ezt egy  $R'$  állandóval jellemezzük, ami az oktaéder középpontja, és egy lap középpontja közötti távolságot jelenti. Ennek kiszámítása:

$$R' = R \sqrt{\frac{\frac{\pi}{12}}{\frac{\sqrt{3}}{4} \operatorname{tg}^2 \left[ \arccos \left( \frac{1}{\sqrt{3}} \right) \right]}}$$

Ezután a segédföldrajzi koordinátákat használjuk. Ha  $\lambda^*$  negatív, vagy nagyobb, mint  $120^\circ$ , addig kell  $120^\circ$ -ot hozzáadni, vagy kivonni, amíg  $0^\circ$  és  $120^\circ$  közé nem esik. A továbbiakban  $\lambda^*$  alatt ezt a korrigált értéket értem. A koordináták kiszámítása néhány paraméter meghatározása után történik:

$$H = \arccos \left( \frac{\sqrt{6}}{6} \sin \lambda^* - \frac{\sqrt{2}}{2} \cos \lambda^* \right)$$

$$A = R^2 \left( \lambda^* + H - \frac{3}{4} \pi \right) \text{ (szögek radiánban értendők)}$$

A polárszög ebből már kiszámítható:

$$\omega = \operatorname{arctg} \left( \frac{2A}{[R']^2 \operatorname{tg}^2 \left[ \arccos \left( \frac{1}{\sqrt{3}} \right) \right] - 2\sqrt{3}A} \right)$$

A polártávolsághoz újabb paraméterekre van szükségünk:

$$q = \operatorname{arctg} \left( \frac{\operatorname{tg} \left[ \arccos \left( \frac{1}{\sqrt{3}} \right) \right]}{\cos \lambda^* + \sqrt{3} \sin \lambda^*} \right)$$

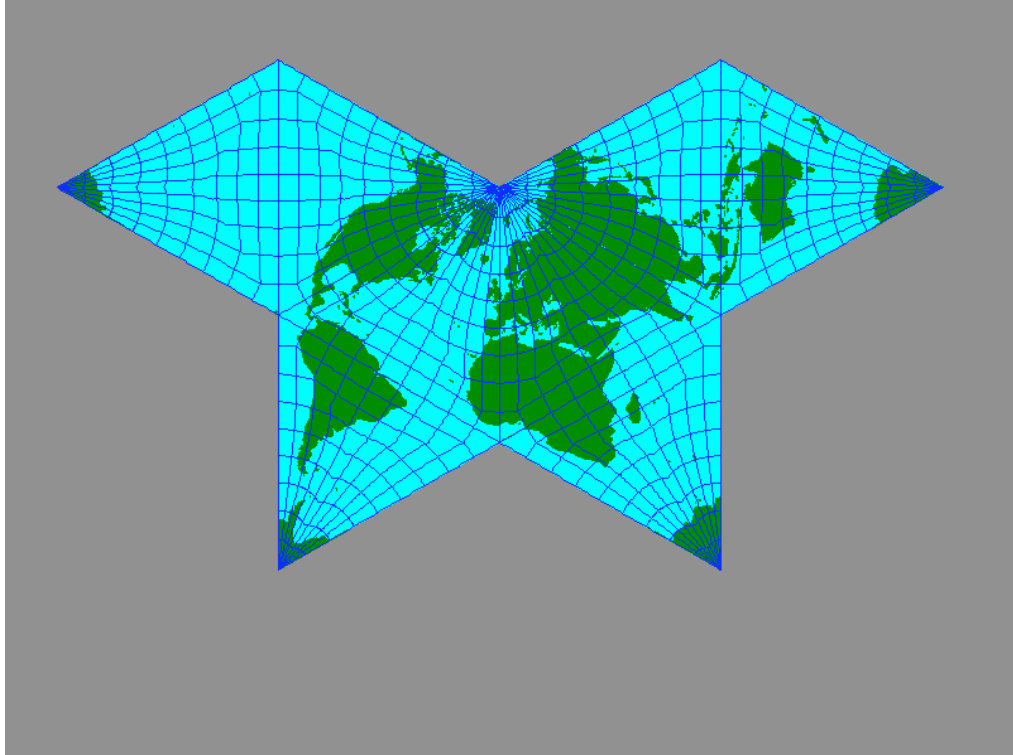
$$d = \frac{R' \operatorname{tg} \left[ \arccos \left( \frac{1}{\sqrt{3}} \right) \right]}{\cos \omega + \sqrt{3} \sin \omega}$$

$$f = \frac{d}{2R' \sin \left( \frac{q}{2} \right)}$$

Ebből kiszámolható:

$$\rho = \frac{2R'}{M} \cdot f \cdot \sin \left( \frac{\frac{\pi}{2} - \varphi^*}{2} \right)$$

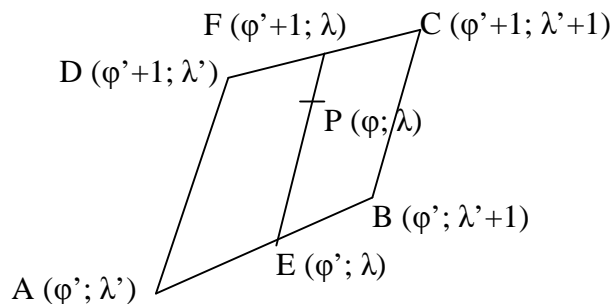
A polárszöghöz végül hozzá kell adni azt a korrekciót, amit a segédhosszúságból az elején kivontunk.



26. ábra: Területtartó Snyder-vetület. Középmeridián:  $0^\circ$ ; fokhálózata  $10^\circ$

### Cahill–Keyes vetület közelítése interpolációval

A Cahill–Keyes vetület egyenlete igen bonyolult, és nagy számításigényű, mert számos különböző zónára osztja a gömbnyolcadot. A gyorsabb számítás és az egyszerűbb program érdekében a vetületet interpolációval közelíttem. Keyes honlapján elérhető egy táblázat, mely a fokhálózati vonalak metszéspontjainak vetületi koordinátáit tartalmazza egész fokokra. Ezek az adatok kilométerben vannak megadva, deciméter pontossággal, a  $0^\circ$ – $90^\circ$  szélesség és a  $0^\circ$ – $45^\circ$  hosszúságok között. Ez csak a gömbnyolcad egy fele, de ha a gömbnyolcad másik felére esik a pont, akkor a pontot először a gömbön a 45. meridiánra tükrözve, majd a képét síkon ugyanezen meridián képére visszatükrözve megkapjuk ezen pontok vetületi koordinátáit is.



27. ábra: Az interpolációhoz felhasznált pontok

## A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

Ezek alapján, ha ismerem egy P pont földrajzi koordinátáit, akkor a táblázatból meg tudom határozni annak az  $1^\circ \times 1^\circ$ -os foktrapéznek a négy sarkát vetületi koordinátákkal, amelyben a pontunk képe lesz. Ezt a foktrapézt egy általános négyszöggel közelítem. A négyszög délnyugati sarkát A-val, a többi az óramutató járásával ellentétes irányban B, C és D betűvel jelölöm. (27. ábra) Az A pont földrajzi koordinátáit, amelyek a P pont földrajzi koordinátáinak egészrészei,  $\varphi'$ -el és  $\lambda'$ -vel jelölöm. Az E és F pontok a P-vel azonos hosszúságon vannak, de az E pont az AB, az F pont a CD szakaszon fekszik. Ezeknek koordinátáit tudom először lineáris interpolációval meghatározni (szögek fokban értendők):

$$x_E = x_A + (x_B - x_A)(\lambda - \lambda')$$

$$y_E = y_A + (y_B - y_A)(\lambda - \lambda')$$

$$x_F = x_D + (x_C - x_D)(\lambda - \lambda')$$

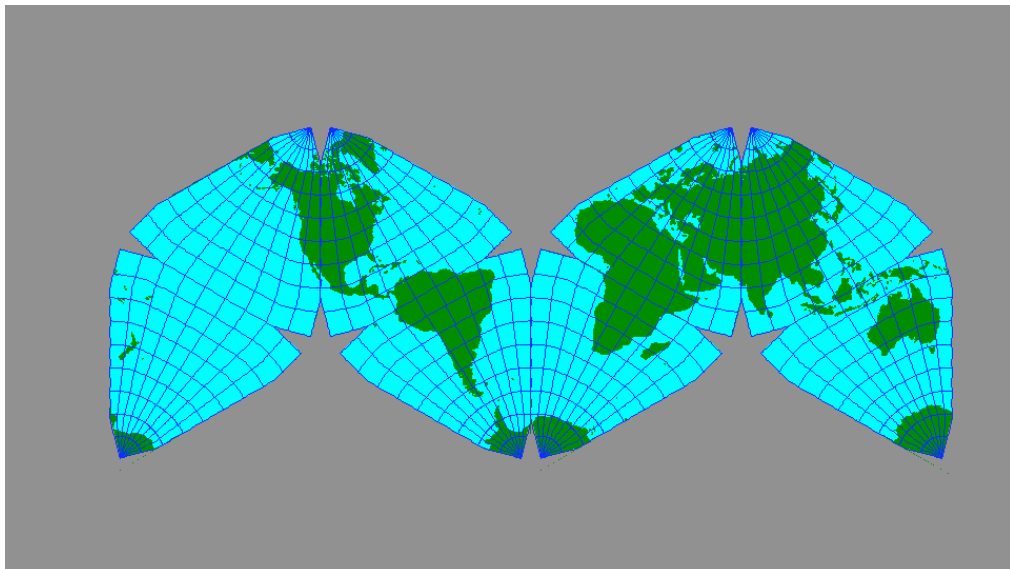
$$y_F = y_D + (y_C - y_D)(\lambda - \lambda')$$

A keresett P pont képét az EF szakaszon végzett újabb interpolációval lehet meghatározni:

$$x = x_E + (x_F - x_E)(\varphi - \varphi')$$

$$y = y_E + (y_F - y_E)(\varphi - \varphi')$$

Keyes délnyugati tájékozású koordináta-rendszert használ. A PHP koordináta-rendszerébe úgy térek át, hogy az y koordináta ellentettjét veszem, majd a két koordinátát megcserélem. ( $90^\circ$ -os elforgatás óra járásával ellentétes irányban.) A méretarány miatt a koordinátákat végül M-mel el kell osztani.



28. ábra: Cahill–Keyes-vetület. Középméridián: Ny. h.  $20^\circ$ ; fokhálózat  $10^\circ$

A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

### Módosított Collignon-vetület

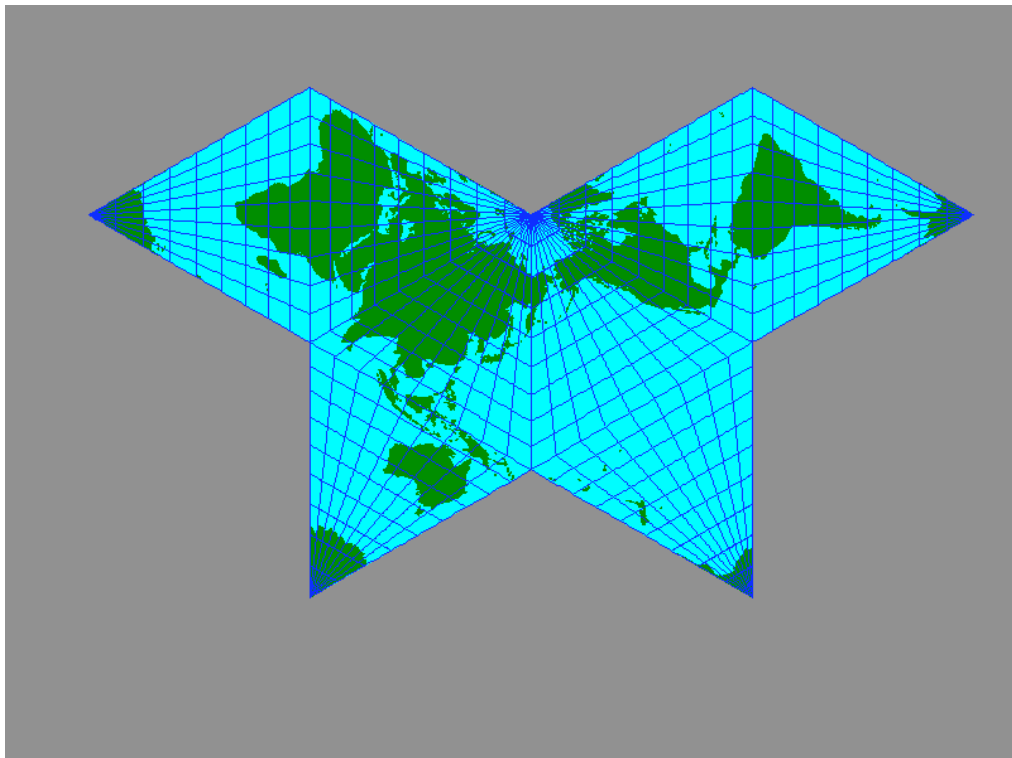
A módosított Collignon-vetülethez nem szükséges segédföldrajzi koordinátákat számítani, viszont  $\Delta\lambda$  alatt egy  $-45^\circ$  és  $45^\circ$  közötti szöget értek, ami a háromszög közepén futó meridián és a leképezendő pont közötti hosszúságkülönbség.

A vetületi egyenletek viszonylag egyszerűek:

$$y = \frac{R}{M} \sqrt{\frac{\sqrt{3}}{2} \pi (1 - \sin \varphi)}$$

$$x = \frac{4}{\sqrt{3}\pi} \Delta\lambda \cdot y \quad (\text{szögek radiánban értendők})$$

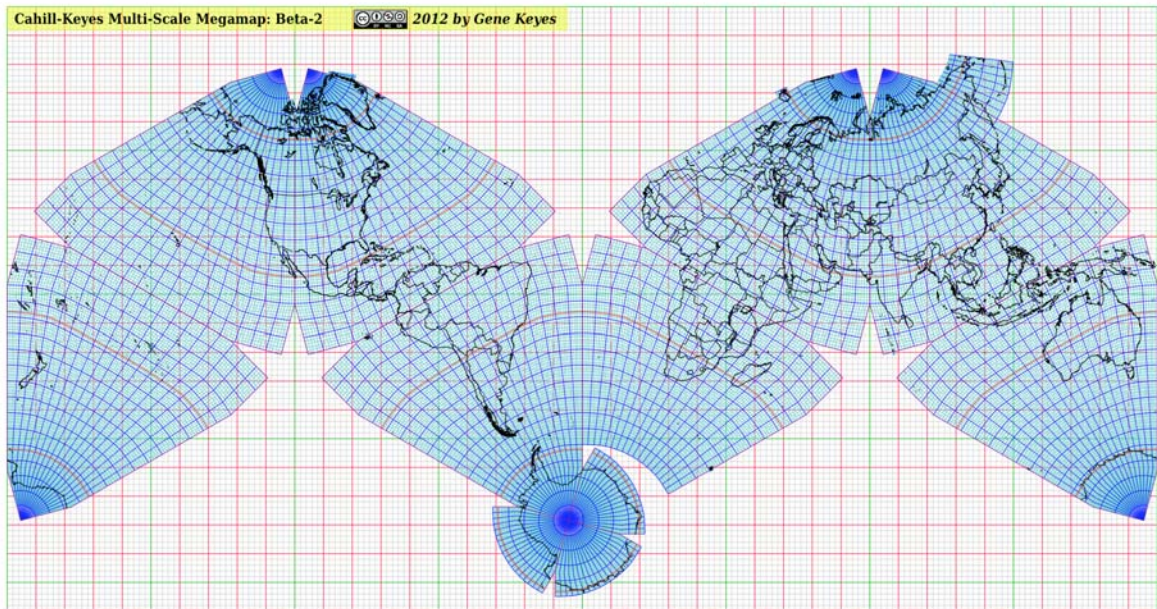
A koordináta-rendszer tájékozása itt szerencsésen egybeesik a PHP koordinátaival.



29. ábra: Módosított Collignon-vetület. Középméridián: K. h.  $160^\circ$ ; fokhálózat  $10^\circ$

## 4. fejezet

### A pillangó-vetületek mai lehetséges alkalmazásai



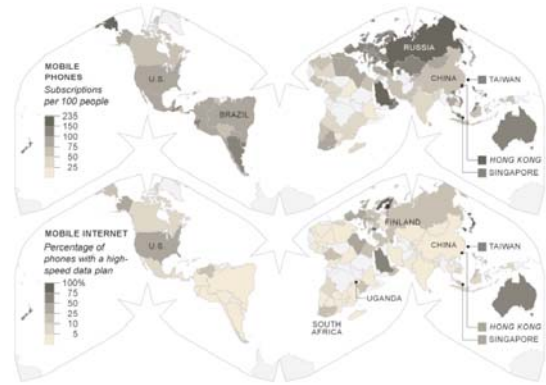
30. ábra: Digitális adatbázisból készült politikai térkép Cahill–Keyes vetületben. Keyes nyomán.

Az oktaéder képfelület egyik nagy előnye, hogy az ilyen képfelületre vetített térképek általában a hagyományos ábrázolásmódoknál kedvezőbb torzulási viszonyokkal rendelkeznek. Cahill eredetileg, amikor a pillangó típusú vetületeit kidolgozta, ezt az előnyt tudományos alkalmazások (pl. meteorológia, navigáció, hírdástechnika) területén szeretne volna kihasználni a gnomonikus és a szögtartó leképezésekkel, és csak a kvázi-területtartó vetületét szánta a nagyközönségnek. A mai GIS technológiák segítségével gyorsabban és pontosabban tudunk szögeket, távolságokat megmérni és a gömbi főköröket megtalálni, mint egy ilyen célú térképen ezeket körzővel meghatározni. Így jelenleg ezeknek a térképeknek inkább esztétikai értékük a fontosabb. Ezt használja ki Waterman, akinek térképeit falitérkép vagy akár hűtőmágnes formátumban az interneten is megvásárolhatjuk. Ő a térképeit oktatási célokra is javasolja, hiszen a kis torzulások miatt a diákok fejében pontosabb kép alakul ki a Földről, mint egy Mercator-vetületű világtérképet szemlélve.

A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben



31. ábra: Hűtőmágnés-térkép  
Waterman-vetületben



32. ábra: Újságban megjelent tematikus  
térkép Cahill–Keyes vetületben

Szintén nem elhanyagolható szempont, amit Clarke fedezett fel, hogy a háromszögeket a négy-fa logika szerint ugyanúgy tovább lehet bontani, mint a hagyományos négyzet alakot, így a geoinformatikában is teret nyerhet alkalmazásuk. Én ezeket a térképeket elsősorban dekorációs (pl. falitérkép) vagy tudományos ismeretterjesztő célokra, tematikus térképek alapjaként ajánlanám. Egy ilyen térkép a nagyközönség figyelmét felhívna magára, hiszen a legtöbb világtérképünk téglalap vagy ellipszis alakú. A szakadások miatt olyan tematika ajánlott, amely csak a kontinensekre terjed ki, az óceánok területére nincs adat. A szögtartó oktaéder vetületet lokális alaktartása és más szögtartó vetületekhez képest csekély torzulásai miatt különösen tudom ajánlani iskolai atlaszokba. A szakdolgozat írása közben (2011. XII. 6-án) jelent meg a New York Times hasábjain néhány tematikus térkép Cahill–Keyes vetületben.

## Összefoglalás

Cahill pillangó-térképei számos más oktaéder vetületet ihlettek. Már maga Cahill is háromféle vetületet hozott létre rendszerében. Ezért sok változatos vetületet használhatunk fel pillangó alakú térkép megalkotásához: Cahill három vetületén (gnomonikus, szögtartó, kvázi-területtartó) kívül a Cahill–Keyes általános torzulású vetület, Snyder és Waterman által megalkotott területtartó vetületek, és kis módosításokkal Collignon területtartó vetülete is alkalmas a képfelületre.

Mivel az oktaéder jobban közelíti a gömböt, mint a sík, a kúp vagy a henger, a hagyományos vetületeknél kedvezőbb torzulási viszonyokat is elérhetünk. Ennek ára a térképen több helyen megjelenő szakadás, de ezen szakadások helyei optimalizálhatóak, hogy az ábrázolás szempontjából kevésbé fontos területeket vágjanak át.

A rendelkezésre álló vetületi egyenletek alapján PHP nyelvű programot írtam. Ez a megadott vetületekben rajzolja ki a fókusz és a földrészek képeit a képernyőre. A vetület középpontját a programhoz tetszőlegesen meg lehet adni.

A pillangó típusú térképeknek ma is nagy szerepe lehetne. Felhasználhatóak oktatási, geoinformatikai vagy pusztán dekoratív célokra, bizonyos tematikus térképeknek kitűnő alapot adnak. Figyelemfelkeltő hatásuk és esztétikai értékük is jelentős.



## **Köszönetnyilvánítás**

A szakdolgozat elkészítésében hatalmas segítséget nyújtott témavezetőm, Dr. Györffy János. A szakdolgozat témáját az ő ötlete nyomán választottam meg. Hálás vagyok neki a heti rendszerességű konzultációkért, és hogy a dolgozat szövegét szakmai és stilisztikai szempontból egyaránt alaposan felülvizsgálta. Rendkívül sokat segített cikkek felkutatásában, a szögtartó vetület megoldásában és a torzulási modulusok kiszámításában, értelmezésében. Köszönöm továbbá tanszékünk lelkiismeretes titkárnőjének, Horváth Ildikónak, hogy a titkárságon található földoktaéder készítőjét alapos munkával kinyomozta. Jelen szakdolgozat nem jöhetett volna létre Dr. Gede Mátyás a PHP nyelv térképészeti alkalmazásait bemutató internetes oktatási segédanyaga nélkül.

## Forrásjegyzék

**Fehér D.:** Poliéder vetületek, *ELTE Térképtudományi és Geoinformatikai Tanszék*, 2003.

**B. J. S. Cahill:** A Land Map of the World on a New Projection, *Journal of the Association of Engineering Societies*, 1913. X. 153-207. oldalak

**B. J. S. Cahill:** A World Map to End World Maps, *Geografiska Annaler*, 16. évf., 97-108. oldalak

**B. J. S. Cahill:** Projection for World Maps, *Monthly Weather Review*, 57. évf., 4. szám, (1929. IV.) 128-133. oldalak

**B. J. S. Cahill:** Map of the World, *szabadalom (sorszám: 1054276)* 1913. II. 25.

**G. Keyes:** Notes on Re-designing B.J.S. Cahill's Butterfly World Map, 2009. XII. 30.

**J. P. Snyder:** An Equal-Area Map Projection for Polyhedral Globes, *Cartographica*, 29. évf., 1. szám, (1992. tavasz) 10-21. oldalak

**J. A. Lugo–K. C. Clarke:** Implementation of Triangulated Quadtree Sequencing for a Global Relief Data Structure, *Twelfth International Symposium on Computer-Assisted Cartography*, 1995. II. 27. 147-156. oldalak

**G. Keyes:** Review of the Waterman World Map, 2010. XI. 24.

**L. P. Lee:** Conformal Projections Based on Elliptic Functions, *Cartographica*, 1976

## Ábrák forrásai

2. ábra: **Madarassy P.–Guszlev A.:** A vetülettan alkalmazási térbeli alakzatokon. A TDK-dolgozat mellékleteként készített oktaédert magam fényképeztem.
3. ábra: **B. J. S. Cahill:** A World Map to End World Maps
4. és 5. ábra: **B. J. S. Cahill:** Projection for World Maps
- 6-8. ábra: **B. J. S. Cahill:** A Land Map of the World on a New Projection
9. ábra **B. J. S. Cahill:** Map of the World
10. ábra: <http://www.genekeyes.com/Cahill-desk-maps/3-Cahill-desk-maps.html>
11. ábra: [http://www.genekeyes.com/CAHILL\\_GALLERY/1975-Cahill-Keyes.html](http://www.genekeyes.com/CAHILL_GALLERY/1975-Cahill-Keyes.html)
- 12-14; 19-22. ábrák: **G. Keyes:** Notes on Re-designing B.J.S. Cahill's Butterfly World Map
15. ábra: **J. P. Snyder:** An Equal-Area Map Projection for Polyhedral Globes
16. ábra: **J. A. Lugo–K. C. Clarke:** Implementation of Triangulated Quadtree Sequencing for a Global Relief Data Structure
17. ábra: <http://www.progonos.com/furuti/MapProj/Dither/ProjPoly/projPoly2.html>
18. és 31. ábrák: <http://odt.org/> webáruház
30. ábra: <http://www.genekeyes.com/CAHILL-KEYES-UPDATE/Cahill-Keyes-map-update2.html>
32. ábra: <http://www.nytimes.com/interactive/2011/12/06/science/1206-world.html>

# 1. melléklet

## Az egyes leképezések összevetése

	gnomonikus	szögtartó	Cahill kvázi-területtartó	Cahill–Keyes ált. torzulású	Snyder területtartó	Collignon területtartó	Waterman területtartó
<b>Zavaróan nagy hossztorzulások</b>	A háromszögek sarkainál	nincs	nincs	nincs	nincs	A pólusoknál	nincs
<b>Nagy törések a fokhalózat vonalában, a földrészek alakja igen torz</b>	A háromszögek határán.	nincs	nincs	A háromszögek csúcsaihoz közel, de ezek tengerre, vagy az Antarktiszra esnek, így nem zavaró.	A háromszögek középpontját a csúcsokkal összekötő vonalak mentén, rendkívül nagy mértékben.	A háromszögek határán, kivéve az Egyenlítő és a 45. meridián találkozását.	A sarkoknál, különösen a 75° szélesség mentén.
<b>Bevágás a háromszögek csúcsain</b>	nincs	nincs	25°, a határ görbe vonalra képeződik le, így nem lehet összeilleszteni.	17°, egy földrészt sem érint, mert Antarktisz össze lehet tolni a térkép alján.	nincs	nincs	kb. 18°, ez Szibériában gondot okoz, Antarktisz viszont a térkép alján egységes pillangó
<b>Készítő által javasolt alak</b>	pillangó	pillangó	pillangó	M	nincs	pillangó	pillangó
<b>Javasolt középmeridiánok</b>	Ny. h. 22°30', K. h. 157°30'	Ny. h. 22°30', K. h. 157°30'	Ny. h. 22°30', K. h. 157°30'	Ny. h. 20°, K. h. 160°	nincs definiálva	nincs definiálva	Ny. h. 20°, K. h. 160°
<b>Készítő által javasolt felhasználás</b>	Gömbi főkörök megtalálása, navigáció	Természet-tudomány, elsősorban meteorológia	Atlasz-kartográfia	Politikai térképek, nagyközönség számára		Geoinformatika, négy-fa eljárás háromszögekre	Oktatás

## 2. melléklet

# A PHP-programok forráskódjai

### index.php

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="hu" lang="hu">
<head>
<title>Oktaéder vetületek</title>
<meta http-equiv="content-type" content="text/html; charset=utf-8" />
<script type="text/javascript">
function rajz()
{
    var meridian = parseFloat(document.getElementById('meridian').value);
    var fok = parseFloat(document.getElementById('fok').value);
    var meret = parseFloat(document.getElementById('meret').value);
    if (meridian > 180 || meridian < -180 || isNaN(meridian))
    {
        alert('Középmeridián érvénytelen!');
    }
    else if (fok > 360 || fok <= 0 || isNaN(fok))
    {
        alert('Fokhálózat érvénytelen!');
    }
    else if (meret <= 0 || isNaN(meret))
    {
        alert('Méretarány érvénytelen!');
    }
    else
    {
        document.getElementById('terkep').src = document.getElementById('vetulet').value + '.php?meridian=' + meridian + '&fok=' + fok + '&meret=' + meret + (document.getElementById('m').checked ? '&m' : '');
    }
}
</script>
</head>
<body>
<h1>Oktaéder vetületek</h1>
<table>
</tr>
<tr>
<td>
<b>Vetület</b>
</td>
<td>
<select id="vetulet">
<option value="keyes">Cahill-Keyes</option>
<option value="snyder">Területtartó (Snyder)</option>
<option value="collignon">Területtartó (Collignon)</option>
<option value="szogt">Szögtartó</option>
<option value="gnomon">Gnomonikus</option>
</select>
</td>
</tr>
<tr>
<td>
<b>Alak</b>
</td>
<td>
<input type="radio" id="pill" name="alak" checked="checked" /> <label for="pill">Pillangó</label>
<input type="radio" id="m" name="alak" /> <label for="m">M alakú</label>
</td>
</tr>
</table>
```

## A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

```
<b>Középmeridián</b><br />
<i>(Valós szám, -180 és 180 között.)</i>
</td>
<td>
<input type="text" id="meridian" size="6" maxlength="6" value="-20" /> °
</td>
</tr>
<tr>
<td>
<b>Fokhálózat sꞑrꞑsége</b><br />
<i>(Csak pozitív valós szám lehet.)</i>
</td>
<td>
<input type="text" id="fok" size="6" maxlength="4" value="10" /> °
</td>
</tr>
<tr>
<td>
<b>Méretarány</b><br />
<i>(Csak pozitív valós szám lehet.)</i>
</td>
<td>
1 képpont = <input type="text" id="meret" size="6" maxlength="4" value="50" /> km.
</td>
</tr>
<tr>
<td>
&nbsp;
</td>
<td>
<input type="button" value="Rajz!" onclick="rajz()" />
</td>
</tr>
</table>
<img src="" alt="" id="terkep" />
</body>
</html>
```

### gnomon.php

```
<?php
$meret = $_GET['meret'];
$fok = $_GET['fok'];
$meridian = $_GET['meridian'];
$kep = isset($_GET['m']) ? imagecreate(60000 / $meret, 33750 / $meret) : imagecreate(60000 / $meret, 45000 / $meret);
$hatter = imagecolorallocate($kep, 127, 127, 127);
$tenger = imagecolorallocate($kep, 0, 255, 255);
$halo = imagecolorallocate($kep, 0, 0, 255);
$fold = imagecolorallocate($kep, 0, 127, 0);
$tol = 6371 * tan(acos(1 / sqrt(3))) / 2;
if(isset($_GET['m']))
{
    haromszog($meridian - 180, 1, (20000 - 2 * $tol) / $meret, 15000 / $meret, -pi() / 6);
    haromszog($meridian - 180, -1, (20000 - 3 * $tol) / $meret, (15000 + sqrt(3) * $tol) / $meret, -pi() / 6);
    haromszog($meridian - 90, 1, 20000 / $meret, 15000 / $meret, pi() / 6);
    haromszog($meridian - 90, -1, (20000 + $tol) / $meret, (15000 + sqrt(3) * $tol) / $meret, pi() / 6);
    haromszog($meridian, 1, (20000 + 4 * $tol) / $meret, 15000 / $meret, -pi() / 6);
    haromszog($meridian, -1, (20000 + 3 * $tol) / $meret, (15000 + sqrt(3) * $tol) / $meret, -pi() / 6);
    haromszog($meridian + 90, 1, (20000 + 6 * $tol) / $meret, 15000 / $meret, pi() / 6);
    haromszog($meridian + 90, -1, (20000 + 7 * $tol) / $meret, (15000 + sqrt(3) * $tol) / $meret, pi() / 6);
}
else
{
    haromszog($meridian - 180, 1, (25000 - $tol) / $meret, (20000 - sqrt(3) * $tol) / $meret, -pi() / 2);
    haromszog($meridian - 180, -1, (25000 - 3 * $tol) / $meret, (20000 - sqrt(3) * $tol) / $meret, -pi() / 2);
    haromszog($meridian - 90, 1, 25000 / $meret, 20000 / $meret, -pi() / 6);
    haromszog($meridian - 90, -1, (25000 - $tol) / $meret, (20000 + sqrt(3) * $tol) / $meret, -pi() / 6);
}
```

## A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

```

haromszog($meridian, 1, (25000 + 2 * $tol) / $meret, 20000 / $meret, pi() / 6);
haromszog($meridian, -
1, (25000 + 3 * $tol) / $meret, (20000 + sqrt(3) * $tol) / $meret, pi() / 6);
haromszog($meridian + 90, 1, (25000 + 3 * $tol) / $meret, (20000 -
sqrt(3) * $tol) / $meret, pi() / 2);
haromszog($meridian + 90, -1, (25000 + 5 * $tol) / $meret, (20000 -
sqrt(3) * $tol) / $meret, pi() / 2);
}
header('Content-type: image/png');
imagepng($kep);
imagedestroy($kep);
function forgat($p, $x, $y, $a)
{
    return array(
        'x' => $p['x'] * cos($a) + $p['y'] * sin($a) + $x,
        'y' => $p['y'] * cos($a) - $p['x'] * sin($a) + $y
    );
}
function vetulet($f, $l, $f0, $l0)
{
    global $meret;
    $f1 = asin(sin($f0) * sin($f) + cos($f0) * cos($f) * cos($l - $l0));
    $b1 = pi() / 2 - $f1;
    $l1 = acos((sin($f) - sin($f0) * sin($f1)) / (cos($f0) * cos($f1)));
    if (sin($l - $l0) > 0 )
    {
        $l1 *= -1;
    }
    else if ($l - $l0 == 0 || $l - $l0 == 180 || $l - $l0 == -180)
    {
        if ($f > $f0)
        {
            $l1 = 0;
        }
        else
        {
            $l1 = pi();
        }
    }
    $r = 6371 * tan($b1) / $meret;
    return array(
        'y' => -$r * cos($l1),
        'x' => -$r * sin($l1)
    );
}
function foldresz($m, $fel, $x, $y, $a)
{
    global $kep, $fold;
    $fajl = file('fold_jav.txt');
    foreach ($fajl as $sor)
    {
        if (trim($sor) == '0 0')
        {
            {
                if (isset($k) && count($k) / 2 >= 3 && isset($siker) && $siker == true)
                {
                    imagefilledpolygon($kep, $k, count($k) / 2, $fold);
                }
                $k = array();
                $el = 0;
                $siker = false;
                $dsark = false;
            }
            else
            {
                $p = explode(' ', trim($sor));
                if ($p[1] - $m > 315)
                {
                    {
                        $p[1] -= 360;
                    }
                }
                else if ($m - $p[1] > 225)
                {
                    {
                        $p[1] += 360;
                    }
                }
                if (((($p[0] >= 0 && $fel == 1) || ($p[0] <= 0 && $fel == -
1)) && ($p[1] >= $m && $p[1] <= $m + 90))
                {
                    $siker = true;
                }
            }
        }
    }
}

```

## A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

```

        $sel = 0;
        $pl = forgat(vetulet(deg2rad($p[0]), deg2rad($p[1]), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
        $k[] = $pl['x'];
        $k[] = $pl['y'];
    }
    else if (((($p[0] >= 0 && $fel == 1) || ($p[0] <= 0 && $fel == -
1)) && ($p[1] >= $m + 90 && $p[1] <= $m + 225))
    {
        if ($p[0] < -60 && (($siker && $sel == -1) || !$siker))
        {
            $sel = 1;
            $siker = true;
            $pl = forgat(vetulet(-pi() / 2, deg2rad($m + 45), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
            $k[] = $pl['x'];
            $k[] = $pl['y'];
        }
        if (!$siker && $sel == -1)
        {
            $k = array();
            $sel = 0;
        }
        if ($sel != -1)
        {
            $sel = 1;
            $pl = forgat(vetulet(deg2rad($p[0]), deg2rad($m + 90), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
            $k[] = $pl['x'];
            $k[] = $pl['y'];
        }
    }
    else if (((($p[0] >= 0 && $fel == 1) || ($p[0] <= 0 && $fel == -
1)) && ($p[1] >= $m - 135 && $p[1] <= $m))
    {
        if ($p[0] < -60 && (($siker && $sel == 1) || !$siker) && !$dsark)
        {
            $sel = -1;
            $siker = true;
            $pl = forgat(vetulet(-pi() / 2, deg2rad($m + 45), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
            $k[] = $pl['x'];
            $k[] = $pl['y'];
        }
        if (!$siker && $sel == 1)
        {
            $k = array();
            $sel = 0;
        }
        if ($sel != 1)
        {
            $sel = -1;
            $pl = forgat(vetulet(deg2rad($p[0]), deg2rad($m), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
            $k[] = $pl['x'];
            $k[] = $pl['y'];
        }
    }
    else if (((($p[0] >= 0 && $fel == -
1) || ($p[0] <= 0 && $fel == 1)) && ($p[1] >= $m && $p[1] <= $m + 90))
    {
        $pl = forgat(vetulet(0, deg2rad($p[1]), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
        $k[] = $pl['x'];
        $k[] = $pl['y'];
    }
}
}
function szelesseg($m, $fel, $x, $y, $a)
{
    global $fok, $kep, $halo;
    for ($f = 0; ($fel == 1 && $f < 90) || ($fel == -1 && $f > -90); $f += $fel * $fok)
    {
        for ($l = $m; $l < $m + 90; $l++)
        {
            $pl = forgat(vetulet(deg2rad($f), deg2rad($l), $fel * (pi() / 2 -

```



## A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

```
acos(1 / sqrt(3)), deg2rad($m + 45)), $x, $y, $a);
    $p2 = forgat(vetulet(deg2rad($f), deg2rad($l + 1), $fel * (pi() / 2 -
acos(1 / sqrt(3)), deg2rad($m + 45)), $x, $y, $a);
    imageline($kep, $p1['x'], $p1['y'], $p2['x'], $p2['y'], $halo);
}
}
}
function hosszusag($m, $fel, $x, $y, $a)
{
    global $fok, $kep, $halo;
    for ($l = ceil($m / $fok) * $fok; $l <= $m + 90; $l += $fok)
    {
        for ($f = 0; ($fel == 1 && $f < 90) || ($fel == -1 && $f > -90); $f += $fel)
        {
            $p1 = forgat(vetulet(deg2rad($f), deg2rad($l), $fel * (pi() / 2 -
acos(1 / sqrt(3)), deg2rad($m + 45)), $x, $y, $a);
            $p2 = forgat(vetulet(deg2rad($f + $fel), deg2rad($l), $fel * (pi() / 2 -
acos(1 / sqrt(3)), deg2rad($m + 45)), $x, $y, $a);
            imageline($kep, $p1['x'], $p1['y'], $p2['x'], $p2['y'], $halo);
        }
    }
    for ($f = 0; ($fel == 1 && $f < 90) || ($fel == -1 && $f > -90); $f += $fel)
    {
        $p1 = forgat(vetulet(deg2rad($f), deg2rad($m), $fel * (pi() / 2 -
acos(1 / sqrt(3)), deg2rad($m + 45)), $x, $y, $a);
        $p2 = forgat(vetulet(deg2rad($f + $fel), deg2rad($m), $fel * (pi() / 2 -
acos(1 / sqrt(3)), deg2rad($m + 45)), $x, $y, $a);
        imageline($kep, $p1['x'], $p1['y'], $p2['x'], $p2['y'], $halo);
    }
    for ($f = 0; ($fel == 1 && $f < 90) || ($fel == -1 && $f > -90); $f += $fel)
    {
        $p1 = forgat(vetulet(deg2rad($f), deg2rad($m + 90), $fel * (pi() / 2 -
acos(1 / sqrt(3)), deg2rad($m + 45)), $x, $y, $a);
        $p2 = forgat(vetulet(deg2rad($f + $fel), deg2rad($m + 90), $fel * (pi() / 2 -
acos(1 / sqrt(3)), deg2rad($m + 45)), $x, $y, $a);
        imageline($kep, $p1['x'], $p1['y'], $p2['x'], $p2['y'], $halo);
    }
}
function haromszog($m, $fel, $x, $y, $a)
{
    global $kep, $tenger;
    $h = array(
        array($fel * pi() / 2, 0),
        array(0, deg2rad($m)),
        array(0, deg2rad($m + 90))
    );
    $h1 = array();
    foreach ($h as $i => $p)
    {
        $p1 = forgat(vetulet($p[0], $p[1], $fel * (pi() / 2 -
acos(1 / sqrt(3)), deg2rad($m + 45)), $x, $y, $a);
        $h1[] = $p1['x'];
        $h1[] = $p1['y'];
    }
    imagefilledpolygon($kep, $h1, 3, $tenger);
    foldresz($m, $fel, $x, $y, $a);
    szelesseg($m, $fel, $x, $y, $a);
    hosszusag($m, $fel, $x, $y, $a);
}
?>
```

## szogt.php

```
<?php
$meret = $_GET['meret'];
$fok = $_GET['fok'];
$meridian = $_GET['meridian'];
$kep = isset($_GET['m']) ? imagecreate(56000 / $meret, 31500 / $meret) : imagecreate(56000 / $
meret, 42000 / $meret);
$shatter = imagecolorallocate($kep, 127, 127, 127);
$tenger = imagecolorallocate($kep, 0, 255, 255);
$halo = imagecolorallocate($kep, 0, 0, 255);
$fold = imagecolorallocate($kep, 0, 127, 0);
$stol = 3979.33130865;
if(isset($_GET['m']))
```

## A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

```

{
  haromszog($meridian - 180, 1, (20000 - 2 * $tol) / $meret, 10000 / $meret, -pi() / 6);
  haromszog($meridian - 180, -1, (20000 -
3 * $tol) / $meret, (10000 + sqrt(3) * $tol) / $meret, -pi() / 6);
  haromszog($meridian - 90, 1, 20000 / $meret, 10000 / $meret, pi() / 6);
  haromszog($meridian - 90, -
1, (20000 + $tol) / $meret, (10000 + sqrt(3) * $tol) / $meret, pi() / 6);
  haromszog($meridian, 1, (20000 + 4 * $tol) / $meret, 10000 / $meret, -pi() / 6);
  haromszog($meridian, -1, (20000 + 3 * $tol) / $meret, (10000 + sqrt(3) * $tol) / $meret, -
pi() / 6);
  haromszog($meridian + 90, 1, (20000 + 6 * $tol) / $meret, 10000 / $meret, pi() / 6);
  haromszog($meridian + 90, -
1, (20000 + 7 * $tol) / $meret, (10000 + sqrt(3) * $tol) / $meret, pi() / 6);
}
else
{
  haromszog($meridian - 180, 1, (25000 - $tol) / $meret, (20000 -
sqrt(3) * $tol) / $meret, -pi() / 2);
  haromszog($meridian - 180, -1, (25000 - 3 * $tol) / $meret, (20000 -
sqrt(3) * $tol) / $meret, -pi() / 2);
  haromszog($meridian - 90, 1, 25000 / $meret, 20000 / $meret, -pi() / 6);
  haromszog($meridian - 90, -1, (25000 -
$tol) / $meret, (20000 + sqrt(3) * $tol) / $meret, -pi() / 6);
  haromszog($meridian, 1, (25000 + 2 * $tol) / $meret, 20000 / $meret, pi() / 6);
  haromszog($meridian, -
1, (25000 + 3 * $tol) / $meret, (20000 + sqrt(3) * $tol) / $meret, pi() / 6);
  haromszog($meridian + 90, 1, (25000 + 3 * $tol) / $meret, (20000 -
sqrt(3) * $tol) / $meret, pi() / 2);
  haromszog($meridian + 90, -1, (25000 + 5 * $tol) / $meret, (20000 -
sqrt(3) * $tol) / $meret, pi() / 2);
}
header('Content-type: image/png');
imagepng($kep);
imagedestroy($kep);
function plusz($a, $b)
{
  return array($a[0] + $b[0], $a[1] + $b[1]);
}
function skalar($a, $l)
{
  return array($l * $a[0], $l * $a[1]);
}
function hatvany($a, $k)
{
  $z = sqrt($a[0] * $a[0] + $a[1] * $a[1]);
  $w = atan2($a[1], $a[0]);
  $z = pow($z, $k);
  $w = $w * $k;
  return array($z * cos($w), $z * sin($w));
}
function sztereookta($r)
{
  $r = skalar($r, 2 * sqrt(2));
  $w = $r;
  $w = plusz($w, skalar(hatvany($r, 4), 0.026042));
  $w = plusz($w, skalar(hatvany($r, 7), 0.003193));
  $w = plusz($w, skalar(hatvany($r, 10), 0.000555));
  $w = plusz($w, skalar(hatvany($r, 13), 0.000113));
  $w = plusz($w, skalar(hatvany($r, 16), 0.000025));
  $w = plusz($w, skalar(hatvany($r, 19), 0.000006));
  $w = plusz($w, skalar(hatvany($r, 22), 0.000002));
  return $w;
}
function forgat($p, $x, $y, $a)
{
  return array(
    'x' => $p['x'] * cos($a) + $p['y'] * sin($a) + $x,
    'y' => $p['y'] * cos($a) - $p['x'] * sin($a) + $y
  );
}
function vetulet($f, $l, $f0, $l0)
{
  global $meret;
  $f1 = asin(sin($f0) * sin($f) + cos($f0) * cos($f) * cos($l - $l0));
  $b1 = pi() / 2 - $f1;
  $l1 = acos((sin($f) - sin($f0) * sin($f1)) / (cos($f0) * cos($f1)));
  if (sin($l - $l0) > 0 )

```

## A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

```

{
    $l1 *= -1;
}
else if ($l - $l0 == 0 || $l - $l0 == 180 || $l - $l0 == -180)
{
    if ($f > $f0)
    {
        $l1 = 0;
    }
    else
    {
        $l1 = pi();
    }
}
if (abs(acos(1 / sqrt(3)) - $b1) < deg2rad(2.5))
{
    $l2 = $l1 - floor($l1 / pi() * 3) * pi() / 3;
    $l2 = $l2 > pi() / 6 ? $l2 - pi() / 3 : $l2;
    $w = $l1 + $l2 / 2;
    $r = 2 * 3979.33130865 * (1 - (acos(1 / sqrt(3)) - $b1) * 3) / $meret;
    return array(
        'x' => -$r * sin($w),
        'y' => -$r * cos($w)
    );
}
else
{
    $r = tan($b1 / 2);
    $fel = $f0 / abs($f0);
    $w = sztereookta(array($fel * $r * cos($l1), $fel * $r * sin($l1)));
    return array(
        'x' => $fel * -6371 * $w[1] / $meret / sqrt(2),
        'y' => $fel * -6371 * $w[0] / $meret / sqrt(2)
    );
}
}
function foldresz($m, $fel, $x, $y, $a)
{
    global $kep, $fold;
    $fajl = file('fold_jav.txt');
    foreach ($fajl as $sor)
    {
        if (trim($sor) == '0 0')
        {
            {
                if (isset($k) && count($k) / 2 >= 3 && isset($siker) && $siker == true)
                {
                    imagefilledpolygon($kep, $k, count($k) / 2, $fold);
                }
                $k = array();
                $el = 0;
                $siker = false;
                $dsark = false;
            }
            else
            {
                $p = explode(' ', trim($sor));
                if ($p[1] - $m > 315)
                {
                    $p[1] -= 360;
                }
                else if ($m - $p[1] > 225)
                {
                    $p[1] += 360;
                }
                if (((($p[0] >= 0 && $fel == 1) || ($p[0] <= 0 && $fel == -
1)) && ($p[1] >= $m && $p[1] <= $m + 90))
                {
                    $siker = true;
                    $el = 0;
                    $p1 = forgat(vetulet(deg2rad($p[0]), deg2rad($p[1]), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
                    $k[] = $p1['x'];
                    $k[] = $p1['y'];
                }
                else if (((($p[0] >= 0 && $fel == 1) || ($p[0] <= 0 && $fel == -
1)) && ($p[1] >= $m + 90 && $p[1] <= $m + 225))
                {

```

## A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

```

        if ($p[0] < -60 && (($siker && $sel == -1) || !$siker))
        {
            $sel = 1;
            $siker = true;
            $p1 = forгат(vetulet(-pi() / 2, deg2rad($m + 45), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
            $k[] = $p1['x'];
            $k[] = $p1['y'];
        }
        if (!$siker && $sel == -1)
        {
            $k = array();
            $sel = 0;
        }
        if ($sel != -1)
        {
            $sel = 1;
            $p1 = forгат(vetulet(deg2rad($p[0]), deg2rad($m + 90), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
            $k[] = $p1['x'];
            $k[] = $p1['y'];
        }
    }
    else if (((($p[0] >= 0 && $fel == 1) || ($p[0] <= 0 && $fel == -
1)) && ($p[1] >= $m - 135 && $p[1] <= $m))
    {
        if ($p[0] < -60 && (($siker && $sel == 1) || !$siker) && !$dsark)
        {
            $sel = -1;
            $siker = true;
            $p1 = forгат(vetulet(-pi() / 2, deg2rad($m + 45), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
            $k[] = $p1['x'];
            $k[] = $p1['y'];
        }
        if (!$siker && $sel == 1)
        {
            $k = array();
            $sel = 0;
        }
        if ($sel != 1)
        {
            $sel = -1;
            $p1 = forгат(vetulet(deg2rad($p[0]), deg2rad($m), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
            $k[] = $p1['x'];
            $k[] = $p1['y'];
        }
    }
    else if (((($p[0] >= 0 && $fel == -
1) || ($p[0] <= 0 && $fel == 1)) && ($p[1] >= $m && $p[1] <= $m + 90))
    {
        $p1 = forгат(vetulet(0, deg2rad($p[1]), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
        $k[] = $p1['x'];
        $k[] = $p1['y'];
    }
}
}
function szelesseg($m, $fel, $x, $y, $a)
{
    global $fok, $kep, $halo;
    for ($f = 0; ($fel == 1 && $f < 90) || ($fel == -1 && $f > -90); $f += $fel * $fok)
    {
        for ($l = $m; $l < $m + 90; $l++)
        {
            $p1 = forгат(vetulet(deg2rad($f), deg2rad($l), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
            $p2 = forгат(vetulet(deg2rad($f), deg2rad($l + 1), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
            imageline($kep, $p1['x'], $p1['y'], $p2['x'], $p2['y'], $halo);
        }
    }
}
function hosszusag($m, $fel, $x, $y, $a)
{

```

## A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

```
global $fok, $kep, $halo;
for ($l = ceil($m / $fok) * $fok; $l <= $m + 90; $l += $fok)
{
    for ($f = 0; ($fel == 1 && $f < 90) || ($fel == -1 && $f > -90); $f += $fel)
    {
        $p1 = forgat(vetulet(deg2rad($f), deg2rad($l), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
        $p2 = forgat(vetulet(deg2rad($f + $fel), deg2rad($l), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
        imageline($kep, $p1['x'], $p1['y'], $p2['x'], $p2['y'], $halo);
    }
}
for ($f = 0; ($fel == 1 && $f < 90) || ($fel == -1 && $f > -90); $f += $fel)
{
    $p1 = forgat(vetulet(deg2rad($f), deg2rad($m), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
    $p2 = forgat(vetulet(deg2rad($f + $fel), deg2rad($m), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
    imageline($kep, $p1['x'], $p1['y'], $p2['x'], $p2['y'], $halo);
}
for ($f = 0; ($fel == 1 && $f < 90) || ($fel == -1 && $f > -90); $f += $fel)
{
    $p1 = forgat(vetulet(deg2rad($f), deg2rad($m + 90), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
    $p2 = forgat(vetulet(deg2rad($f + $fel), deg2rad($m + 90), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
    imageline($kep, $p1['x'], $p1['y'], $p2['x'], $p2['y'], $halo);
}
}
function haromszog($m, $fel, $x, $y, $a)
{
    global $kep, $tenger;
    $h = array(
        array($fel * pi() / 2, 0),
        array(0, deg2rad($m)),
        array(0, deg2rad($m + 90))
    );
    $h1 = array();
    foreach ($h as $i => $p)
    {
        $p1 = forgat(vetulet($p[0], $p[1], $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
        $h1[] = $p1['x'];
        $h1[] = $p1['y'];
    }
    imagefilledpolygon($kep, $h1, 3, $tenger);
    foldresz($m, $fel, $x, $y, $a);
    szelesseg($m, $fel, $x, $y, $a);
    hosszusag($m, $fel, $x, $y, $a);
}
?>
```

## snyder.php

```
<?php
$meret = $_GET['meret'];
$fok = $_GET['fok'];
$meridian = $_GET['meridian'];
$kep = isset($_GET['m']) ? imagecreate(48000 / $meret, 27000 / $meret) : imagecreate(48000 / $
meret, 36000 / $meret);
$hatter = imagecolorallocate($kep, 127, 127, 127);
$tenger = imagecolorallocate($kep, 0, 255, 255);
$halo = imagecolorallocate($kep, 0, 0, 255);
$fold = imagecolorallocate($kep, 0, 127, 0);
$tol = 6371 * sqrt(sqrt(3) / 2 * pi()) / 3;
if(isset($_GET['m']))
{
    haromszog($meridian - 180, 1, (15000 - 2 * $tol) / $meret, 10000 / $meret, -pi() / 6);
    haromszog($meridian - 180, -1, (15000 -
3 * $tol) / $meret, (10000 + sqrt(3) * $tol) / $meret, -pi() / 6);
    haromszog($meridian - 90, 1, 15000 / $meret, 10000 / $meret, pi() / 6);
    haromszog($meridian - 90, -
1, (15000 + $tol) / $meret, (10000 + sqrt(3) * $tol) / $meret, pi() / 6);
    haromszog($meridian, 1, (15000 + 4 * $tol) / $meret, 10000 / $meret, -pi() / 6);
    haromszog($meridian, -1, (15000 + 3 * $tol) / $meret, (10000 + sqrt(3) * $tol) / $meret, -
```

## A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

```

pi() / 6);
    haromszog($meridian + 90, 1, (15000 + 6 * $tol) / $meret, 10000 / $meret, pi() / 6);
    haromszog($meridian + 90, -
1, (15000 + 7 * $tol) / $meret, (10000 + sqrt(3) * $tol) / $meret, pi() / 6);
}
else
{
    haromszog($meridian - 180, 1, (20000 - $tol) / $meret, (15000 -
sqrt(3) * $tol) / $meret, -pi() / 2);
    haromszog($meridian - 180, -1, (20000 - 3 * $tol) / $meret, (15000 -
sqrt(3) * $tol) / $meret, -pi() / 2);
    haromszog($meridian - 90, 1, 20000 / $meret, 15000 / $meret, -pi() / 6);
    haromszog($meridian - 90, -1, (20000 -
$tol) / $meret, (15000 + sqrt(3) * $tol) / $meret, -pi() / 6);
    haromszog($meridian, 1, (20000 + 2 * $tol) / $meret, 15000 / $meret, pi() / 6);
    haromszog($meridian, -
1, (20000 + 3 * $tol) / $meret, (15000 + sqrt(3) * $tol) / $meret, pi() / 6);
    haromszog($meridian + 90, 1, (20000 + 3 * $tol) / $meret, (15000 -
sqrt(3) * $tol) / $meret, pi() / 2);
    haromszog($meridian + 90, -1, (20000 + 5 * $tol) / $meret, (15000 -
sqrt(3) * $tol) / $meret, pi() / 2);
}
header('Content-type: image/png');
imagepng($kep);
imagedestroy($kep);

function forgat($p, $x, $y, $a)
{
    return array(
        'x' => $p['x'] * cos($a) + $p['y'] * sin($a) + $x,
        'y' => $p['y'] * cos($a) - $p['x'] * sin($a) + $y
    );
}

function vetulet($f, $l, $f0, $l0)
{
    global $meret;
    $f1 = asin(sin($f0) * sin($f) + cos($f0) * cos($f) * cos($l - $l0));
    $b1 = pi() / 2 - $f1;
    $l1 = acos((sin($f) - sin($f0) * sin($f1)) / (cos($f0) * cos($f1)));
    if (sin($l - $l0) > 0)
    {
        $l1 *= -1;
    }
    else if ($l - $l0 == 0 || $l - $l0 == 180 || $l - $l0 == -180)
    {
        if ($f > $f0)
        {
            $l1 = 0;
        }
        else
        {
            $l1 = pi();
        }
    }
    $l2 = $l1;
    if ($f0 < 0)
    {
        $l2 += pi();
    }
    while ($l2 < 0 || $l2 > 2 * pi() / 3)
    {
        if ($l2 > 0)
        {
            $l2 -= 2 * pi() / 3;
        }
        else
        {
            $l2 += 2 * pi() / 3;
        }
    }
    $r1 = sqrt((pi() / 12) / (0.5 * tan(acos(1 / sqrt(3))) * tan(acos(1 / sqrt(3))) * sqrt(3)
/ 4));
    $h = acos(sin($l2) * sqrt(2) / 2 * cos(acos(1 / sqrt(3))) - cos($l2) * sqrt(2) / 2);
    $ag = $l2 + pi() / 4 + $h - pi();
    $w = atan2(2 * $ag, $r1 * $r1 * tan(acos(1 / sqrt(3))) * tan(acos(1 / sqrt(3))) -
2 * $ag * sqrt(3));
    $q = atan(tan(acos(1 / sqrt(3))) / (cos($l2) + sin($l2) * sqrt(3)));

```

## A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

```

$d = $r1 * tan(acos(1 / sqrt(3))) / (cos($w) + sin($w) * sqrt(3));
$f2 = $d / (2 * $r1 * sin($q / 2));
$w += $l1 - $l2;
$r = 2 * $r1 * $f2 * sin($b1 / 2) * 6371 / $meret;
return array(
    'y' => -$r * cos($w),
    'x' => -$r * sin($w)
);
}
function foldresz($m, $fel, $x, $y, $a)
{
    global $kep, $fold;
    $fajl = file('fold_jav.txt');
    foreach ($fajl as $sor)
    {
        if (trim($sor) == '0 0')
        {
            if (isset($k) && count($k) / 2 >= 3 && isset($siker) && $siker == true)
            {
                imagefilledpolygon($kep, $k, count($k) / 2, $fold);
            }
            $k = array();
            $sel = 0;
            $siker = false;
            $dsark = false;
        }
        else
        {
            $p = explode(' ', trim($sor));
            if ($p[1] - $m > 315)
            {
                $p[1] -= 360;
            }
            else if ($m - $p[1] > 225)
            {
                $p[1] += 360;
            }
            if ((($p[0] >= 0 && $fel == 1) || ($p[0] <= 0 && $fel == -
1)) && ($p[1] >= $m && $p[1] <= $m + 90))
            {
                $siker = true;
                $sel = 0;
                $p1 = forgat(vetulet(deg2rad($p[0]), deg2rad($p[1]), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
                $k[] = $p1['x'];
                $k[] = $p1['y'];
            }
            else if ((($p[0] >= 0 && $fel == 1) || ($p[0] <= 0 && $fel == -
1)) && ($p[1] >= $m + 90 && $p[1] <= $m + 225))
            {
                if ($p[0] < -60 && (($siker && $sel == -1) || !$siker))
                {
                    $sel = 1;
                    $siker = true;
                    $p1 = forgat(vetulet(-pi() / 2, deg2rad($m + 45), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
                    $k[] = $p1['x'];
                    $k[] = $p1['y'];
                }
                if (!$siker && $sel == -1)
                {
                    $k = array();
                    $sel = 0;
                }
                if ($sel != -1)
                {
                    $sel = 1;
                    $p1 = forgat(vetulet(deg2rad($p[0]), deg2rad($m + 90), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
                    $k[] = $p1['x'];
                    $k[] = $p1['y'];
                }
            }
            else if ((($p[0] >= 0 && $fel == 1) || ($p[0] <= 0 && $fel == -
1)) && ($p[1] >= $m - 135 && $p[1] <= $m))
            {
                if ($p[0] < -60 && (($siker && $sel == 1) || !$siker) && !$dsark)

```

## A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

```

        {
            $sel = -1;
            $siker = true;
            $p1 = forgat(vetulet(-pi() / 2, deg2rad($m + 45), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
            $k[] = $p1['x'];
            $k[] = $p1['y'];
        }
        if (!$siker && $sel == 1)
        {
            $k = array();
            $sel = 0;
        }
        if ($sel != 1)
        {
            $sel = -1;
            $p1 = forgat(vetulet(deg2rad($p[0]), deg2rad($m), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
            $k[] = $p1['x'];
            $k[] = $p1['y'];
        }
    }
    else if (($p[0] >= 0 && $fel == -
1) || ($p[0] <= 0 && $fel == 1)) && ($p[1] >= $m && $p[1] <= $m + 90)
    {
        $p1 = forgat(vetulet(0, deg2rad($p[1]), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
        $k[] = $p1['x'];
        $k[] = $p1['y'];
    }
}
}
function szelesseg($m, $fel, $x, $y, $a)
{
    global $fok, $kep, $halo;
    for ($f = 0; ($fel == 1 && $f < 90) || ($fel == -1 && $f > -90); $f += $fel * $fok)
    {
        for ($l = $m; $l < $m + 90; $l++)
        {
            $p1 = forgat(vetulet(deg2rad($f), deg2rad($l), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
            $p2 = forgat(vetulet(deg2rad($f), deg2rad($l + 1), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
            imageline($kep, $p1['x'], $p1['y'], $p2['x'], $p2['y'], $halo);
        }
    }
}
function hosszusag($m, $fel, $x, $y, $a)
{
    global $fok, $kep, $halo;
    for ($l = ceil($m / $fok) * $fok; $l <= $m + 90; $l += $fok)
    {
        for ($f = 0; ($fel == 1 && $f < 90) || ($fel == -1 && $f > -90); $f += $fel)
        {
            $p1 = forgat(vetulet(deg2rad($f), deg2rad($l), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
            $p2 = forgat(vetulet(deg2rad($f + $fel), deg2rad($l), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
            imageline($kep, $p1['x'], $p1['y'], $p2['x'], $p2['y'], $halo);
        }
        for ($f = 0; ($fel == 1 && $f < 90) || ($fel == -1 && $f > -90); $f += $fel)
        {
            $p1 = forgat(vetulet(deg2rad($f), deg2rad($m), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
            $p2 = forgat(vetulet(deg2rad($f + $fel), deg2rad($m), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
            imageline($kep, $p1['x'], $p1['y'], $p2['x'], $p2['y'], $halo);
        }
        for ($f = 0; ($fel == 1 && $f < 90) || ($fel == -1 && $f > -90); $f += $fel)
        {
            $p1 = forgat(vetulet(deg2rad($f), deg2rad($m + 90), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
            $p2 = forgat(vetulet(deg2rad($f + $fel), deg2rad($m + 90), $fel * (pi() / 2 -
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);
            imageline($kep, $p1['x'], $p1['y'], $p2['x'], $p2['y'], $halo);
        }
    }
}

```



## A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

```
    }  
}  
function haromszog($m, $fel, $x, $y, $a)  
{  
    global $kep, $tenger;  
    $h = array(  
        array($fel * pi() / 2, 0),  
        array(0, deg2rad($m)),  
        array(0, deg2rad($m + 90))  
    );  
    $h1 = array();  
    foreach ($h as $i => $p)  
    {  
        $p1 = forгат(vetulet($p[0], $p[1], $fel * (pi() / 2 -  
acos(1 / sqrt(3))), deg2rad($m + 45)), $x, $y, $a);  
        $h1[] = $p1['x'];  
        $h1[] = $p1['y'];  
    }  
    imagefilledpolygon($kep, $h1, 3, $tenger);  
    foldresz($m, $fel, $x, $y, $a);  
    szelesseg($m, $fel, $x, $y, $a);  
    hosszusag($m, $fel, $x, $y, $a);  
}  
>>
```

### keyes.php

```
<?php  
$meret = $_GET['meret'];  
$fok = $_GET['fok'];  
$meridian = $_GET['meridian'];  
$kx = file('keyesx.csv');  
$ky = file('keyesy.csv');  
foreach ($kx as $id => $sor)  
{  
    $kx[$id] = explode(',', $sor);  
}  
foreach ($ky as $id => $sor)  
{  
    $ky[$id] = explode(',', $sor);  
}  
$kep = isset($_GET['m']) ? imagecreate(48000 / $meret, 27000 / $meret) : imagecreate(40000 / $  
meret, 30000 / $meret);  
$hatter = imagecolorallocate($kep, 127, 127, 127);  
$tenger = imagecolorallocate($kep, 0, 255, 255);  
$halo = imagecolorallocate($kep, 0, 0, 255);  
$fold = imagecolorallocate($kep, 0, 127, 0);  
$tol = 10000;  
if(isset($_GET['m']))  
{  
    haromszog($meridian - 180, 1, 15000 / $meret, 5000 / $meret, -pi() / 6);  
    haromszog($meridian - 180, -1, (15000 -  
$tol) / $meret, (5000 + sqrt(3) * $tol) / $meret, -pi() / 6);  
    haromszog($meridian - 90, 1, 15000 / $meret, 5000 / $meret, pi() / 6);  
    haromszog($meridian - 90, -  
1, (15000 + $tol) / $meret, (5000 + sqrt(3) * $tol) / $meret, pi() / 6);  
    haromszog($meridian, 1, (15000 + 2 * $tol) / $meret, 5000 / $meret, -pi() / 6);  
    haromszog($meridian, -1, (15000 + $tol) / $meret, (5000 + sqrt(3) * $tol) / $meret, -  
pi() / 6);  
    haromszog($meridian + 90, 1, (15000 + 2 * $tol) / $meret, 5000 / $meret, pi() / 6);  
    haromszog($meridian + 90, -  
1, (15000 + 3 * $tol) / $meret, (5000 + sqrt(3) * $tol) / $meret, pi() / 6);  
}  
else  
{  
    haromszog($meridian - 180, 1, 20000 / $meret, 10000 / $meret, -pi() / 2);  
    haromszog($meridian - 180, -1, (20000 - 2 * $tol) / $meret, 10000 / $meret, -pi() / 2);  
    haromszog($meridian - 90, 1, 20000 / $meret, 10000 / $meret, -pi() / 6);  
    haromszog($meridian - 90, -1, (20000 -  
$tol) / $meret, (10000 + $tol * sqrt(3)) / $meret, -pi() / 6);  
    haromszog($meridian, 1, 20000 / $meret, 10000 / $meret, pi() / 6);  
    haromszog($meridian, -  
1, (20000 + $tol) / $meret, (10000 + $tol * sqrt(3)) / $meret, pi() / 6);  
    haromszog($meridian + 90, 1, 20000 / $meret, 10000 / $meret, pi() / 2);  
    haromszog($meridian + 90, -1, (20000 + 2 * $tol) / $meret, 10000 / $meret, pi() / 2);  
}
```

## A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

```

}
header('Content-type: image/png');
imagepng($kep);
imagedestroy($kep);

function forgat($p, $x, $y, $a)
{
    return array(
        'x' => $p['x'] * cos($a) + $p['y'] * sin($a) + $x,
        'y' => $p['y'] * cos($a) - $p['x'] * sin($a) + $y
    );
}

function vetulet($f, $l, $f0, $l0)
{
    global $meret, $kx, $ky;
    $l1 = abs($l - $l0);
    $f1 = abs($f);
    $xa = 1 * $kx[floor($f1)][floor($l1)];
    $ya = 1 * $ky[floor($f1)][floor($l1)];
    $xb = 1 * $kx[floor($f1)][floor($l1) + 1];
    $yb = 1 * $ky[floor($f1)][floor($l1) + 1];
    $xc = 1 * $kx[floor($f1) + 1][floor($l1) + 1];
    $yc = 1 * $ky[floor($f1) + 1][floor($l1) + 1];
    $xd = 1 * $kx[floor($f1) + 1][floor($l1)];
    $yd = 1 * $ky[floor($f1) + 1][floor($l1)];
    $xe = $xa + ($xb - $xa) * ($l1 - floor($l1));
    $ye = $ya + ($yb - $ya) * ($l1 - floor($l1));
    $xf = $xd + ($xc - $xd) * ($l1 - floor($l1));
    $yf = $yd + ($yc - $yd) * ($l1 - floor($l1));
    $x = $xe + ($xf - $xe) * ($f1 - floor($f1));
    $y = $ye + ($yf - $ye) * ($f1 - floor($f1));
    if (($l - $l0) < 0)
    {
        $y *= -1;
    }
    if (($f0) < 0)
    {
        $x *= -1;
    }
    return array(
        'y' => $x / $meret,
        'x' => $y / $meret
    );
}

function foldresz($m, $fel, $x, $y, $a)
{
    global $kep, $fold;
    $fajl = file('fold_jav.txt');
    foreach ($fajl as $sor)
    {
        if (trim($sor) == '0 0')
        {
            if (isset($k) && count($k) / 2 >= 3 && isset($siker) && $siker == true)
            {
                imagefilledpolygon($kep, $k, count($k) / 2, $fold);
            }
            $k = array();
            $sel = 0;
            $siker = false;
            $dsark = false;
        }
        else
        {
            $p = explode(' ', trim($sor));
            if ($p[1] - $m > 315)
            {
                $p[1] -= 360;
            }
            else if ($m - $p[1] > 225)
            {
                $p[1] += 360;
            }
            if (((($p[0] >= 0 && $fel == 1) || ($p[0] <= 0 && $fel == -
1)) && ($p[1] >= $m && $p[1] <= $m + 90))
            {
                $siker = true;
                $sel = 0;
            }
        }
    }
}

```

## A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

```

        $p1 = forgat(vetulet($p[0], $p[1], $fel, $m + 45), $x, $y, $a);
        $k[] = $p1['x'];
        $k[] = $p1['y'];
    }
    else if (((($p[0] >= 0 && $fel == 1) || ($p[0] <= 0 && $fel == -
1)) && ($p[1] >= $m + 90 && $p[1] <= $m + 225))
    {
        if ($p[0] < -60 && (($siker && $sel == -1) || !$siker))
        {
            $sel = 1;
            $siker = true;
            $p1 = forgat(vetulet(-90, $m - 1, $fel, $m + 45), $x, $y, $a);
            $k[] = $p1['x'];
            $k[] = $p1['y'];
        }
        if (!$siker && $sel == -1)
        {
            $k = array();
            $sel = 0;
        }
        if ($sel != -1)
        {
            $sel = 1;
            $p1 = forgat(vetulet($p[0], $m + 90, $fel, $m + 45), $x, $y, $a);
            $k[] = $p1['x'];
            $k[] = $p1['y'];
        }
    }
    else if (((($p[0] >= 0 && $fel == 1) || ($p[0] <= 0 && $fel == -
1)) && ($p[1] >= $m - 135 && $p[1] <= $m))
    {
        if ($p[0] < -60 && (($siker && $sel == 1) || !$siker) && !$dsark)
        {
            $sel = -1;
            $siker = true;
            $p1 = forgat(vetulet(-90, $m - 1, $fel, $m + 45), $x, $y, $a);
            $k[] = $p1['x'];
            $k[] = $p1['y'];
        }
        if (!$siker && $sel == 1)
        {
            $k = array();
            $sel = 0;
        }
        if ($sel != 1)
        {
            $sel = -1;
            $p1 = forgat(vetulet($p[0], $m, $fel, $m + 45), $x, $y, $a);
            $k[] = $p1['x'];
            $k[] = $p1['y'];
        }
    }
    else if (((($p[0] >= 0 && $fel == -
1) || ($p[0] <= 0 && $fel == 1)) && ($p[1] >= $m && $p[1] <= $m + 90))
    {
        $p1 = forgat(vetulet(0, $p[1], $fel, $m + 45), $x, $y, $a);
        $k[] = $p1['x'];
        $k[] = $p1['y'];
    }
}
}
}
function szelesseg($m, $fel, $x, $y, $a)
{
    global $fok, $kep, $halo;
    for ($f = 0; ($fel == 1 && $f < 90) || ($fel == -1 && $f > -90); $f += $fel * $fok)
    {
        for ($l = $m; $l < $m + 90; $l ++)
        {
            $p1 = forgat(vetulet($f, $l, $fel, $m + 45), $x, $y, $a);
            $p2 = forgat(vetulet($f, $l + 1, $fel, $m + 45), $x, $y, $a);
            imageline($kep, $p1['x'], $p1['y'], $p2['x'], $p2['y'], $halo);
        }
    }
}
function hosszusag($m, $fel, $x, $y, $a)
{

```

## A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

```
global $fok, $kep, $halo;
for ($l = ceil($m / $fok) * $fok; $l <= $m + 90; $l += $fok)
{
    for ($f = 0; ($fel == 1 && $f < 90) || ($fel == -1 && $f > -90); $f += $fel)
    {
        $p1 = forgat(vetulet($f, $l, $fel, $m + 45), $x, $y, $a);
        $p2 = forgat(vetulet($f + $fel, $l, $fel, $m + 45), $x, $y, $a);
        imageline($kep, $p1['x'], $p1['y'], $p2['x'], $p2['y'], $halo);
    }
}
for ($f = 0; ($fel == 1 && $f < 90) || ($fel == -1 && $f > -90); $f += $fel)
{
    $p1 = forgat(vetulet($f, $m, $fel, $m + 45), $x, $y, $a);
    $p2 = forgat(vetulet($f + $fel, $m, $fel, $m + 45), $x, $y, $a);
    imageline($kep, $p1['x'], $p1['y'], $p2['x'], $p2['y'], $halo);
}
for ($f = 0; ($fel == 1 && $f < 90) || ($fel == -1 && $f > -90); $f += $fel)
{
    $p1 = forgat(vetulet($f, $m + 90, $fel, $m + 45), $x, $y, $a);
    $p2 = forgat(vetulet($f + $fel, $m + 90, $fel, $m + 45), $x, $y, $a);
    imageline($kep, $p1['x'], $p1['y'], $p2['x'], $p2['y'], $halo);
}
}
function haromszog($m, $fel, $x, $y, $a)
{
    global $kep, $tenger;
    $h = array(
        array($fel * 90, $m),
        array($fel * 73, $m),
        array($fel * 15, $m),
        array(0, $m),
        array(0, $m + 16),
        array(0, $m + 74),
        array(0, $m + 90),
        array($fel * 15, $m + 90),
        array($fel * 73, $m + 90)
    );
    $h1 = array();
    foreach ($h as $i => $p)
    {
        $p1 = forgat(vetulet($p[0], $p[1], $fel, $m + 45), $x, $y, $a);
        $h1[] = $p1['x'];
        $h1[] = $p1['y'];
    }
    imagefilledpolygon($kep, $h1, 9, $tenger);
    foldresz($m, $fel, $x, $y, $a);
    $h = array(
        array($fel * 90, $m),
        array($fel * 73, $m),
        array($fel * 90, $m - 1),
        array($fel * 73, $m + 90)
    );
    $h1 = array();
    foreach ($h as $i => $p)
    {
        $p1 = forgat(vetulet($p[0], $p[1], $fel, $m + 45), $x, $y, $a);
        $h1[] = $p1['x'];
        $h1[] = $p1['y'];
    }
    imagefilledpolygon($kep, $h1, 4, $tatter);
    szelesseg($m, $fel, $x, $y, $a);
    hosszusag($m, $fel, $x, $y, $a);
}
?>
```

## collignon.php

```
<?php
$meret = $_GET['meret'];
$fok = $_GET['fok'];
$meridian = $_GET['meridian'];
$kep = isset($_GET['m']) ? imagecreate(48000 / $meret, 27000 / $meret) : imagecreate(48000 / $meret, 36000 / $meret);
$hatter = imagecolorallocate($kep, 127, 127, 127);
$tenger = imagecolorallocate($kep, 0, 255, 255);
```

## A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

```

$halo = imagecolorallocate($kep, 0, 0, 255);
$fold = imagecolorallocate($kep, 0, 127, 0);
$stol = 6371 * sqrt(sqrt(3) / 2 * pi());
if(isset($_GET['m']))
{
    haromszog($meridian - 180, 1, 15000 / $meret, 5000 / $meret, -pi() / 6);
    haromszog($meridian - 180, -1, (15000 -
    $stol) / $meret, (5000 + sqrt(3) * $stol) / $meret, -pi() / 6);
    haromszog($meridian - 90, 1, 15000 / $meret, 5000 / $meret, pi() / 6);
    haromszog($meridian - 90, -
    1, (15000 + $stol) / $meret, (5000 + sqrt(3) * $stol) / $meret, pi() / 6);
    haromszog($meridian, 1, (15000 + 2 * $stol) / $meret, 5000 / $meret, -pi() / 6);
    haromszog($meridian, -1, (15000 + $stol) / $meret, (5000 + sqrt(3) * $stol) / $meret, -
    pi() / 6);
    haromszog($meridian + 90, 1, (15000 + 2 * $stol) / $meret, 5000 / $meret, pi() / 6);
    haromszog($meridian + 90, -
    1, (15000 + 3 * $stol) / $meret, (5000 + sqrt(3) * $stol) / $meret, pi() / 6);
}
else
{
    haromszog($meridian - 180, 1, 25000 / $meret, 10000 / $meret, -pi() / 2);
    haromszog($meridian - 180, -1, (25000 - 2 * $stol) / $meret, 10000 / $meret, -pi() / 2);
    haromszog($meridian - 90, 1, 25000 / $meret, 10000 / $meret, -pi() / 6);
    haromszog($meridian - 90, -1, (25000 -
    $stol) / $meret, (10000 + $stol * sqrt(3)) / $meret, -pi() / 6);
    haromszog($meridian, 1, 25000 / $meret, 10000 / $meret, pi() / 6);
    haromszog($meridian, -
    1, (25000 + $stol) / $meret, (10000 + $stol * sqrt(3)) / $meret, pi() / 6);
    haromszog($meridian + 90, 1, 25000 / $meret, 10000 / $meret, pi() / 2);
    haromszog($meridian + 90, -1, (25000 + 2 * $stol) / $meret, 10000 / $meret, pi() / 2);
}
header('Content-type: image/png');
imagepng($kep);
imagedestroy($kep);
function forgat($p, $x, $y, $a)
{
    return array(
        'x' => $p['x'] * cos($a) + $p['y'] * sin($a) + $x,
        'y' => $p['y'] * cos($a) - $p['x'] * sin($a) + $y
    );
}
function vetulet($f, $l, $f0, $l0)
{
    global $meret;
    $f1 = $f0 * $f;
    $l1 = $l - $l0;
    while ($l1 < -pi() / 2 || $l1 > pi() / 2)
    {
        if ($l1 > pi() / 4)
        {
            $l1 -= pi();
        }
        else
        {
            $l1 += pi();
        }
    }
    $y = 6371 / $meret * sqrt(sqrt(3) / 2 * pi() * (1 - sin($f1)));
    $x = 4 / sqrt(3) / pi() * $l1 * $y;
    return array(
        'y' => $f0 * $y,
        'x' => $x
    );
}
function foldresz($m, $fel, $x, $y, $a)
{
    global $kep, $fold;
    $fajl = file('fold_jav.txt');
    foreach ($fajl as $sor)
    {
        if (trim($sor) == '0 0')
        {
            if (isset($k) && count($k) / 2 >= 3 && isset($siker) && $siker == true)
            {
                imagefilledpolygon($kep, $k, count($k) / 2, $fold);
            }
            $k = array();
        }
    }
}

```

## A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

```

        $sel = 0;
        $siker = false;
        $dsark = false;
    }
    else
    {
        $p = explode(' ', trim($sor));
        if ($p[1] - $m > 315)
        {
            $p[1] -= 360;
        }
        else if ($m - $p[1] > 225)
        {
            $p[1] += 360;
        }
        if (((($p[0] >= 0 && $fel == 1) || ($p[0] <= 0 && $fel == -
1)) && ($p[1] >= $m && $p[1] <= $m + 90))
        {
            $siker = true;
            $sel = 0;
            $p1 = forgat(vetulet(deg2rad($p[0]), deg2rad($p[1]), $fel, deg2rad($m + 45)),
$x, $y, $a);
            $k[] = $p1['x'];
            $k[] = $p1['y'];
        }
        else if (((($p[0] >= 0 && $fel == 1) || ($p[0] <= 0 && $fel == -
1)) && ($p[1] >= $m + 90 && $p[1] <= $m + 225))
        {
            if ($p[0] < -60 && (($siker && $sel == -1) || !$siker))
            {
                $sel = 1;
                $siker = true;
                $p1 = forgat(vetulet(-
pi() / 2, deg2rad($m + 45), $fel, deg2rad($m + 45)), $x, $y, $a);
                $k[] = $p1['x'];
                $k[] = $p1['y'];
            }
            if (!$siker && $sel == -1)
            {
                $k = array();
                $sel = 0;
            }
            if ($sel != -1)
            {
                $sel = 1;
                $p1 = forgat(vetulet(deg2rad($p[0]), deg2rad($m + 90), $fel, deg2rad($m +
45)), $x, $y, $a);
                $k[] = $p1['x'];
                $k[] = $p1['y'];
            }
        }
        else if (((($p[0] >= 0 && $fel == 1) || ($p[0] <= 0 && $fel == -
1)) && ($p[1] >= $m - 135 && $p[1] <= $m))
        {
            if ($p[0] < -60 && (($siker && $sel == 1) || !$siker) && !$dsark)
            {
                $sel = -1;
                $siker = true;
                $p1 = forgat(vetulet(-
pi() / 2, deg2rad($m + 45), $fel, deg2rad($m + 45)), $x, $y, $a);
                $k[] = $p1['x'];
                $k[] = $p1['y'];
            }
            if (!$siker && $sel == 1)
            {
                $k = array();
                $sel = 0;
            }
            if ($sel != 1)
            {
                $sel = -1;
                $p1 = forgat(vetulet(deg2rad($p[0]), deg2rad($m), $fel, deg2rad($m + 45)),
$x, $y, $a);
                $k[] = $p1['x'];
                $k[] = $p1['y'];
            }
        }
    }
}

```

## A Föld geokartográfiai ábrázolása pillangó típusú oktaéder vetületekben

```

        else if ((($p[0] >= 0 && $fel == -
1) || ($p[0] <= 0 && $fel == 1)) && ($p[1] >= $m && $p[1] <= $m + 90))
    {
        $p1 = forgat(vetulet(0, deg2rad($p[1]), $fel, deg2rad($m + 45)), $x, $y, $a);
        $k[] = $p1['x'];
        $k[] = $p1['y'];
    }
}
}
function szelesseg($m, $fel, $x, $y, $a)
{
    global $fok, $kep, $halo;
    for ($f = 0; ($fel == 1 && $f < 90) || ($fel == -1 && $f > -90); $f += $fel * $fok)
    {
        for ($l = $m; $l < $m + 90; $l ++)
        {
            $p1 = forgat(vetulet(deg2rad($f), deg2rad($l), $fel, deg2rad($m + 45)), $x, $y, $a
);
            $p2 = forgat(vetulet(deg2rad($f), deg2rad($l + 1), $fel, deg2rad($m + 45)), $x, $y
, $a);
            imageline($kep, $p1['x'], $p1['y'], $p2['x'], $p2['y'], $halo);
        }
    }
}
function hosszusag($m, $fel, $x, $y, $a)
{
    global $fok, $kep, $halo;
    for ($l = ceil($m / $fok) * $fok; $l <= $m + 90; $l += $fok)
    {
        for ($f = 0; ($fel == 1 && $f < 90) || ($fel == -1 && $f > -90); $f += $fel)
        {
            $p1 = forgat(vetulet(deg2rad($f), deg2rad($l), $fel, deg2rad($m + 45)), $x, $y, $a
);
            $p2 = forgat(vetulet(deg2rad($f + $fel), deg2rad($l), $fel, deg2rad($m + 45)), $x,
$y, $a);
            imageline($kep, $p1['x'], $p1['y'], $p2['x'], $p2['y'], $halo);
        }
    }
    for ($f = 0; ($fel == 1 && $f < 90) || ($fel == -1 && $f > -90); $f += $fel)
    {
        $p1 = forgat(vetulet(deg2rad($f), deg2rad($m), $fel, deg2rad($m + 45)), $x, $y, $a);
        $p2 = forgat(vetulet(deg2rad($f + $fel), deg2rad($m), $fel, deg2rad($m + 45)), $x, $y,
$a);
        imageline($kep, $p1['x'], $p1['y'], $p2['x'], $p2['y'], $halo);
    }
    for ($f = 0; ($fel == 1 && $f < 90) || ($fel == -1 && $f > -90); $f += $fel)
    {
        $p1 = forgat(vetulet(deg2rad($f), deg2rad($m + 90), $fel, deg2rad($m + 45)), $x, $y, $
a);
        $p2 = forgat(vetulet(deg2rad($f + $fel), deg2rad($m + 90), $fel, deg2rad($m + 45)), $x
, $y, $a);
        imageline($kep, $p1['x'], $p1['y'], $p2['x'], $p2['y'], $halo);
    }
}
function haromszog($m, $fel, $x, $y, $a)
{
    global $kep, $tenger;
    $h = array(
        array($fel * pi() / 2, 0),
        array(0, deg2rad($m)),
        array(0, deg2rad($m + 90))
    );
    $h1 = array();
    foreach ($h as $i => $p)
    {
        $p1 = forgat(vetulet($p[0], $p[1], $fel, deg2rad($m + 45)), $x, $y, $a);
        $h1[] = $p1['x'];
        $h1[] = $p1['y'];
    }
    imagefilledpolygon($kep, $h1, 3, $tenger);
    foldresz($m, $fel, $x, $y, $a);
    szelesseg($m, $fel, $x, $y, $a);
    hosszusag($m, $fel, $x, $y, $a);
}
?>

```

## Nyilatkozat

Alulírott, Kerkovits Krisztián András nyilatkozom, hogy jelen szakdolgozatom teljes egészében saját, önálló szellemi termékem. A szakdolgozatot sem részben, sem egészében semmilyen más felsőfokú oktatási vagy egyéb intézménybe nem nyújtottam be. A szakdolgozatomban felhasznált, szerzői joggal védett anyagokra vonatkozó engedély a mellékletben megtalálható.

A témavezető által benyújtásra elfogadott szakdolgozat PDF formátumban való elektronikus publikálásához a tanszéki honlapon

HOZZÁJÁRULOK

NEM JÁRULOK HOZZÁ

Budapest, 2012. május 15.

.....  
a hallgató aláírása