ABOUT THE PAGE-RANKING ALGORITHM.
A web-lap fontosságának megállapítási módjáról.

Dr. Gyarmati Péter
prof. emeritus

Annyi évvel a kidolgozás, és a bevezetés, sikeres bevezetés, óta időszerű, hogy áttekintsük mi is ez a sikeres rendszer, amely meghódította szinte a teljes 3w-t (www, world-wide-web, világháló, internetnek is mondjuk) és előidézője több más eredménynek. Az alapgondolat, mint okozó, ma is érvényes, nevezetesen az akármennyire is megnövekvő hálózat, egyenjogú és személytelen maradjon, legyen független szervezetektől, kormányoktól, stb.

Legyen tehát egy olyan rendszere az embereknek, az egész világra kiterjedően, amelyben bárki tetszőlegesen információkat szerezhet be –már ami a hálózaton van- és amelyen az ember a józan emberi mivoltából fakadóan tetszőleges információt közölhet. Tudom ez az óhaj számos nyitott kérdést vet fel, azonban engedtessék meg, csak itt és most, ez a szabadosság a teljes jószándék feltételezésével. Meglátjuk, az eredmények igazolni fogják.

A 3w, mint logikai, virtuális hálózat egy bizonyos értelemben egységes –mindenki által érthető, értelmezhető- szerkezetű írásokból, a weblapokból, vagy röviden **lap**okból, és az ezeket elérni szándékozók kereséséből származó elérésekből, **hivatkozás**okból áll.

A fizikai rendszer természetesen tartalmaz a valódi hardware hálózatban számos egyéb segédeszközt, a különböző célú kiszolgálókat, stb. ezek azonban akkor működnek jól, ha a mi rendszerünkből észrevétlenek maradnak!

Nyilvánvaló, hogy a különböző lapok más és más információkat tartalmaznak és más a hozzáférésük. Egyeseket többször, gyakrabban kersenek fel, mint másokat. A lapoknak ezáltal különböző fontosságot tulajdoníthatunk: kimondhatjuk, egy lap annál fontosabb a hálózat szempontjából, minél több a rá való hivatkozás. Tehát egy keresésnél a fontosabb lapokhoz fordulunk először, hiszen már bizonyították, hogy a keresett információ inkább rajtuk található! Ezt az elvet kívánjuk megvalósítani! Nézzük hogyan!

Miért is kell egyáltalán, miért is lényeges ilyen megkülönböztetést tenni?

A hálózatban rengeteg lap van, például a 3w-n milliárdos nagyságrendben, ha minden keresésnél végig kellene menünk a lapokon, amig meg nem találjuk a kívánt információt tartalmazót, akkor nagyon lassú és ezáltal használhatatlan lenne a hálózat. Ennek feloldására számos megoldás készült, nem egy az emberek sokéves gyakorlatából származtatja magát: például eszünkbe sem jut alma vásárlásakor a ruhaboltba menni. Mit is teszünk tulajdonképen? Felteszünk egy rendező elvet, amely mentén fogjuk a keresést végezni, ezáltal lerövidítjük a müveletet. A fenti példában szakcsoportokat képeztünk és a keresésben először azt állapítjuk meg, majd csak azok között „lapozunk”. Jó megoldás, de például a multidiszciplináris esetek itt határesetek, nem eldönthetőek.

Számos más megoldás is elképzelhető, sőt más-más hálózatokon ma is alkalmaznak különböző technikákat. Mi most egy olyan algoritmust szeretnénk bemutatni, amely az eredményét tekintve nem túl bonyolult eljárással meghatározható, adaptiv a változó hálózatra, képes tetszőleges nagyságú hálózatban a keresést optimalizálni, de legalább elfogadható sebességen tartani.

Ha tudnánk, hogy az egyes információkra mennyire van szüksége a keresőknek, vagyis, ha tunánk az információk ilyen értelmű értékét, akkor csak annyit kellene tennünk, hogy az információk helyeit eszerint sorba rakjuk. Sajnos minden próbálkozás erre nézve sikertelen és előbb-utóbb szaknévsorokká válnak: a könyvtárak, az információs központok, a telefon és más kommunikációs elérések, mind valahogyan téma szerint csoportosulnak és azon belül abc és más nem logikus sorrendekben találhatóak. Ráadásul ezeknek még az is feltétele, hogy az egyes témacsoportok megegyezzenek abban, hogy mit ajánlanak (ha mindkettő adja nem nagy baj, de, ha egyik sem!). Látjuk, tudjuk, mindennap találkozunk ezekkel!

Vajon lehetséges-e olyan megoldást találni, ahol a szolgáltatók függetlenek egymástól, ahol a keresést nem korlátozza a hálózat tömege, amely a változó igényekhez adaptáható, amely egyaránt hasznos a tartalmi keresés esetében is.

Nos erre született a 3w hálózat gigantikus méretéhez a bevezetőben elmondott gondolat a **Stanfordi Egyetem**en. A hálózatban a weblapokat jellemezhetjük a rá való hivatkozásokkal! Ez az adat, akár ismerjük értékét, akár nem, létezik a hálózatban! Ha feltesszük, hogy ezt ki tudjuk számítani és eszerint a lapokat „sorba állítani”, akkor megoldottuk a feladatot, hiszen nyertünk egy olyan keresést, amely pontosan a hálózaton nyüzsgők kivánsága szerint működik! Ráadásul, ha ezt a sorrendet időnként felülvizsgáljuk, akkor eleget teszünk az időben változó követelményeknek is! Továbbá, az egyes lapok teljesen függetlenek egymástól, azaz semmiféle megállapodásra, információ elosztásra nincs szükség. A hálózat méretére sem tettünk semmiféle kikötést!

Tehát a feladat ennek a hálózatban létező számnak a meghatározása!

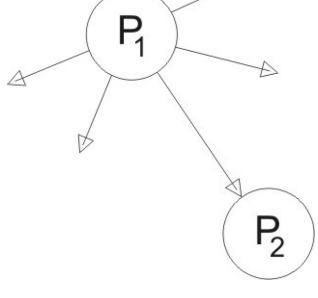
Legelőször nevezzük el: legyen ennek a mérőszámnak a neve **lap fontossági szám**, annak a mértéke, hogy a környezetében lévő oldalak hányszor hivatkoztak rá. A hivatkozott lap szempontjából sokszor visszafelé hivatkozásnak fogjuk hívni.

Minél nagyobb ez a szám, annál fontosabb az oldal, annál nagyobb a valószínűsége, hogy a keresett információ ott található. Ha a sikeres találatok számát rögzítjük és eszerint újrarendezzük a keresési sorrendet, akkor az adott helyzetre pontosíthatjuk, aktualizáljuk a keresést.

Szükségünk lesz tehát egy olyan rekurziv eljárásra, algoritmusra, amely ezt a lapfontosságot előállítja a hálózat tejes egészére, minden lapra. A fontosság kezelésének és a keresés, vagyis a kulcsszavak és kifejezések kapcsolatainak technikájáról most nem beszélünk. Feladatunk „csupán” annak a matematikai módszernek a feltárása, amelynek segítségével a lapfontosság mérhető, megállapítható tetszőleges nagyságú és struktúrájú hálózatban.

A hosszú bevezető után térjünk a lényegre és nézzük először, hogyan is néz ki a fontosság a hálózat saját szerkezetében:

Legyen egy hálózat egy irányított gráf az alábbi értelemben:



1. a gráf pontja a hálózat egy lapja;
2. a gráf éle hivatkozás egy lapra, más lapról, minden lapról külön él fut minden külön lapra hivatkozáshoz.
3. a gráf él iránya a hivatkozás iránya, a nyíl a hivatkozott lapra mutat.
4. A P_1 lapnak k_1 számú hivatkozása van különböző lapokra, köztük egy a P_2-re.

Ilyenkor azt mondjuk, hogy a P_1 $\frac{1}{k_1}$ fontosságot ad P_2-nek!

Definició:

Legyen P_i a hálózat egy lapja, amelyhez feltételezzük, hogy csatlakozik a $P_1; P_2; ...; P_n$ lap, rendre $k_1; k_2; ...; k_n$ csatlakozással, akkor a P_i fontossága:

$f_i = \frac{f_1}{k_1} + \frac{f_2}{k_2} + ... + \frac{f_n}{k_n}$, vagyis a P_i lap fontossága a ráhivatkozó lapok fontosságának összege!

Tulajdonságok.

1. A lap tevékenysége *invariáns* a saját fontosságára nézve.
2. A lap fontossága *visszamenőleges* szám, azaz az őt keresők határozzák meg.
3. Egy „új” hálózatban a lapok fontossága *definiálatlan* és a hivatkozások hozzák létre, azok útján terjed szét a hálózatra! A fontosságnak csak a már működő hálózatban van értelme (említettük, hogy a hálózatban *létező mérőszám*ról van szó).
4. Új belépő lap, vagy nem hivatkozott lap fontossága nem megállapítható, *definiálatlan*!
5. *Reguláris*nak nevezzük azt a hálózatot, amelynek bármely lapjáról el lehet jutni bármely másik lapjára.
6. *Tisztá*nak nevezzük azt a hálózatot, amelyben minden lapnak van hivatkozása, vagyis létezik a fontossági szám.
7. *Monogám*nak nevezzük azt a hálózatot, amelynek egy kereső rendszere van.

Kapcsolati mátrix.

Egy tetszőleges hálózatot leírhatunk táblázatos formában, ahol a sorfejek és az oszlopfejek rendre a hálózat lapjai, míg a tábla belseje a lapok közötti súlyozott kapcsolatot tartalmazza: egy n elemű hálózat esetében n sorból és n oszlopból álló táblázatot kapunk.

P	*1*	*2*	*3*	*...*	*i*	*...*
1						
2						
3						
...						
j					$1/k_j$	
...						

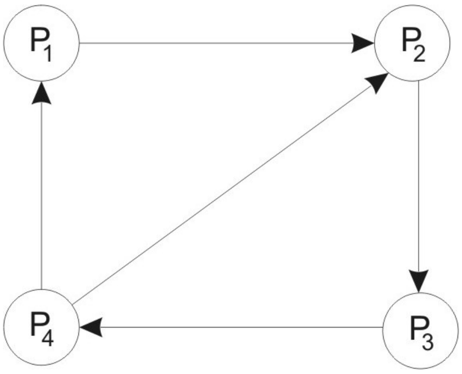
Az i-edik oszlop és j-edik sor kereszteződésében lévő tábla elem a P_i lapnak P_j laptól való hivatkozására vonatkozik:

$=0$, ha P_j-nek nincs hivatkozása P_i-hez;

$= 1/k_j$, ha van hivatkozás és k_j a P_j összes hivatkozása.

A táblázat tartalmából négyzetes mátrixot képezhetünk, amely pontosan egy n elemű hálózat kapcsolati rendszerét írja le, ezért **kapcsolati mátrixnak** nevezzük.

Egy példán mutatjuk be ezt a hálózatleírási technikát. Legyen egy hálózat az alábbi:



A kpcsolati mátrix így néz ki:

P	*1*	*2*	*3*	*4*
1	*0*	*1/2*	*0*	*0*
2	*0*	*0*	*1*	*0*
3	*0*	*0*	*0*	*1*
4	*1*	*1/2*	*0*	*0*

Például a második elem az első sorban ½, mert P_2-re van hivatkozás P_1-ről és az összes P_2-re hivatkozás $k_2=2$. Akkor a mátrix:

$$K=\begin{bmatrix} 0 & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & \frac{1}{2} & 0 & 0 \end{bmatrix}$$

Láthatjuk, hogy az oszlopok összege mindenütt 1, mivel az, a súlyozott fontosságok összege.

Fontossági vektor.

Tegyük fel, hogy ismerjük minden egyes lap fontossági mértékét: a $P_i \Rightarrow f_i$, akkor az összeset sorbagyűjtve a teljes hálózat fontosságát kapjuk:

$F=(f_1; f_2; \ldots;f_i;\ldots;f_n)$

Az F egy n elemű vektor és a **hálózat fontossági vektor**ának nevezzük.

Ha most visszatérünk egy pillanatra a kapcsolati mátrixhoz, akkor ennek analógiájára megállapíthatjuk, hogy a mátrix sorai rendre az egyes lapok fontossági sorrend vektorai.. Azt is mondhatnánk, hogy a teljes mátrix a teljes hálózat fontossági mátrixa!

A feladat tehát az $F=(f_1; f_2; \ldots;f_i;\ldots;f_n)$ vektor kiszámítása, azaz az egyes P_i lapok f_i fontossága, amely szerint a keresési sorrendet meghatározzuk. A legfontosabb lap tartalmazza a legtöbb keresett információt, hiszen így állítottuk fel a fontossági kritériumot!

Amennyiben a hálózat változik –a keresések átmennek jellemzően más lapokra- kapcsolati mátrixot és a fontossági vektort újra elő kell állítani! Ennek módszere másik feladat.

Nézzük most, hogyan tudjuk meghatározni a fontossági vektort?

A feladatot már részben elvégeztük az előbbiekben egy példa kapcsán azáltal, hogy felvételeztük a kapcsolati mátrixot! Ugyanis, ha feltesszük, hogy ismerjük az F fontossági vektort, akkor $KxF=F$ azonosságnak fenn kell állnia a hálózat méretétől összetételétől függetlenül, annak értelmében, ahogy azokat definiáltuk.

Hogyan?

1. A K mátrixunk nxn elemű;
2. Az F vektor n elemű;
3. KxF szorzat is n elemű;
4. Hogyan keletkezik KxF?

- a K mátrix **első sor**ának elemeit rendre összeszorozzuk az **F** vektor elemeivel (elsőt az elsővel, másodikat a másodikkal, ... , az utolsót az utolsóval), majd ezeket összeadjuk: ez lesz a szorzat első eleme;
- a fenti műveletet elvégezzük a K mátrix **második sor**ával: ez lesz a szorzat második eleme;
- a műveleteket az összes sorral elvégezzük: így megkapjuk a szorzatvektort.
- mellesleg ezt a műveletet a lineáris algebrában *lineáris kombináció*nak nevezzük.

Most már csak azt kell belátnunk, hogy ez a szorzatvektor azonos az F vektorral.

Emlékezzünk, hogyan építettük fel a K mátrixot: az első sor a P_1 lapra vonatkozó hivatkozásokat tartalmazza: -ha az adott P_i lapról nincs hivatkozás, akkor nulla, egyébként $1/k_i$; ahol ki a P_i lap összes hivatkozása. Tehát az első elem 0, vagy $1/k_1$; a második 0, vagy $1/k_2$;...; az utolsó 0, vagy $1/k_n$.

Akkor a szorzatvektor első eleme 0, vagy f_1/k_1; a második 0, vagy f_2/k_2;...; az utolsó 0, vagy f_n/k_n.

Ha ezeket összegezzük, akkor azt látjuk, hogy a szorzatvektor első eleme az összege azon fontosságoknak, amelyet a rá hivatkatkozó oldalaktól „kapott", vagyis a saját fontossága! Tehát nem más, mint f_1, ami az F vektor első eleme!

Az összegzést megismételve minden elemre, azt látjuk, hogy a szorzat-vektor éppen F, tehát $KxF=F$.

Matematikai értelmezésben az ilyen tulajdonságú vektort a ***mátrix sajátvektor***ának nevezzük.

Pontosan azt mondjuk, hogy egy V vektor az M mátrix sajátvektora s sajátértékkel, ha az MxV szorzat eredménye az sV vektor (a skalár s szorzás azt jelenti, hogy a vektor minden eleme meg van szorozva s-értékével). Ha a sajátérték $s=1$, akkor $MxV=V$, minden más esetben s-el való súlyozásról beszélünk.

Megjegyzendő, hogy a zérus vektor a fenti feltételeket mindig kielégíti, hiszen $0xM=0$, de semmit nem mond a hálózatról ezért érdektelen a számunkra. Tehát a feladat mindig **a nem-nulla sajátvektor meghatározása**!

A példánkbban feltételeztük, hogy s=1 és a mátrixalgebra felhasználásával

F=(1/7; 2/7; 2/7; 2/7)   adódott.

Értékeljük a vektort egy pillanatra: az a meglepő eredmény látszik, hogy az első lap kevésbé fontos, mint az összes többi, amelyek azonos fontoságúak! Akárhogyan nézegetjük a hálózatunkat ez nem látszik rajta, úgy mondanánk nem-triviális! Tehát már jól jártunk, mert kiderítettünk olyan hálózati tulajdonságot, amelyről nem tudtunk, nem is sejtettük! (ne feledjük a lap fontossága jellemzően „nem saját tulajdonság”, ezért lehetségesek meglepő eredmények).

Manapság a hálózatok elég nagyok, sok eleműek, például a $3w\ \ 10^9$ nagyságrendű, ezért egy ilyen mátrix sajátvektorainak kiszámítása nem kis feladat. Ezért a gyakorlatban egyéb trükkökhöz és közelítő számításokhoz folyamodnak. Ez legyen a programozók feladata, az elmélettel azonban még korántsem végeztünk!

Az elv, amit az eddigiekben felállítottunk több dolgot feltételez a hálózatról, amit matematikailag, mint az elmélet tulajdonságai felsoroltunk: eszerint a módszerünk **a relativ statikus, reguláris, tiszta hálózatok fontossági értékeinek meghatározására** alkalmas. A fentiek mellett még azt is fel kell tennünk, hogy a hálózatunk **monogám**, azaz csak egy, a saját fontosság-mérő rendszerével, kereső rendszerével üzemel.

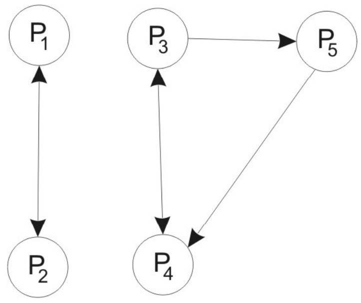
Fordítsuk le ezt a hálózat nyelvére:

1. **Statikus.** A hálózat „lassan” változik: a kapcsolati mátrix változása a fontossági vektort befolyásolja. A kérdés, hogy „milyen változás” esetén szükséges az újraszámolás, vagyis a fontossági sorrend újbóli megállapítása.
2. **Reguláris.** A hálózat bármely lapjáról el lehet jutni bármely lapjára a hivatkozások mentén. Ez a reguláris hálózat! A valóságban mindig vannak olyan csoportok, szigetek, amelyeknek nincs hivatkozása csoporton kivűli lapokkal. Azt is mondhatnánk, hogy ezek önálló hálózatok.
3. **Tiszta.** A hálózat minden lapjára van hivatkozás. A valóságban számos olyan lap is létezhet, amelyik részt vesz a hálózaton keresésben, azonban őrá nincs hivatkozás! Mivel ezek kapcsolati mátrix elemei mind nullák, a fontossága sem állapítható meg, pedig akármilyen aktivitással részt vehetnek a hálózatban!
4. **Monogám.** A hálózat minden hivatkozása ebben a rendszerben történik. A valóságban a lapokat másik kereső rendszerekből és közvetlen eléréssel is használják!

Nézzük meg hogyan segíthetünk ezeken a „bajokon”:

1. **Nem- statikus**. Elméletünk nincs rá, de szerencsére a valóságos hálózatok nem sietik el a dolgukat, ami persze várható is tőlük. A kisebb változások programtechnikai módszerekkel elvégezhetők a keresők találati valószínűségei alapján. Ha nagy kiesések vannak, akkor bizony vannak sebességi problémák, amig a helyettes lap „bejáratódik”, azaz a helyettesállító „ugyanolyan paraméterekkel” be tudja üzemelni a pótlást. A 3w például a Google-nál –nem-hivatalos adatok szerint havonta egyszer futtatja újra a kapcsolati mátrixot.
2. **Nem reguláris** hálózatoknál elméleti trükkhöz folyamodunk, amelyet az alábbi példával mutatunk be.

   Legyen egy hálózat adott az alábbi módon:



A kapcsolati mátrixa: $K=\begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{2} & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{2} & 0 \end{bmatrix}$

Ennek két sajátvektora van, amelyek nem egymás többszörösei, például a *(1;1;0;0;0)* és a *(0;0;2;2;0).* (Ha több megoldás van, akkor nem egyértelmű a dolog, ahogy azt vártuk is).

Ha a mátrixot két sub-mátrixra bontjuk, ahogy az a valós hálózaton is van, akkor mindkét mátrixnak van sajátvektora 1 sajátértékkel:

$$K_1=\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix};\ \text{és}\quad K_2=\begin{bmatrix} 0 & \frac{1}{2} & 1 \\ 1 & 0 & 0 \\ 0 & \frac{1}{2} & 0 \end{bmatrix};$$

akkor $F_1=(1;\ 1)$ *és* $F_2=(2;\ 2;\ 1)$

Ha kiegészítjük nullákkal a teljes elemszámra, rendre visszakapjuk az eredeti mátrix két sajátvektorát, tehát nem követtünk el hibát.

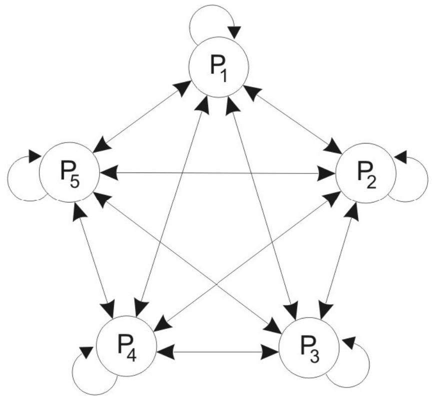
Lehetséges lenne tehát két független hálózatként kezelni, azonban ez egyrészt nem elegáns, másrészt bármely pillanatban megváltozhat a dolog és a két sub összekötődik.

Ha nem tudsz mást, hát folyamodj trükkökhöz: íme egy ilyen trükk, amit a Google alkalmaz.

Alkossunk egy új B mátrixot, amelyet az eredetivel összekombinálunk az alábbi módon:

$(1-m)K+mB$, valamely $m=0$ *és* 1 közötti számra.

ahol K az eredeti, míg B egy olyan kapcsolati mátrix, ahol minden kapcsolódik mindennel, önmagával is és pont annyi eleme van, mint K-nak. Talán így bonyolultnak tűnik, rajzoljuk fel, hát a legutóbbi példánkhoz a B mátrixot:



K elemszáma 5, tehát ez is annyi és minden lapról él megy minden másiklapra és önmagára is, vagyis ennek a kapcsolati mátrixában minden egyes elem $1/5$.

Alkalmazzuk a fenti lineáris kombinációt:

-a K minden elemét megszorozzuk $(1-m)$-el, míg a B-t m-el, majd összeadjuk rendre a kettő elemeit, képezve a kombinációs mátrixot. A Google nem adja közre csodaszámait, de azok előbb-utóbb kitudódnak: a mi példánkon az $m=0,15$ értékkel számolunk, mint a legutóbb kitudódottal. Íme a kombinációs mátrix:

Az eredmény: $\begin{bmatrix} 0,03 & 0,88 & 0,03 & 0,03 & 0,03 \\ 0,88 & 0,03 & 0,03 & 0,03 & 0,03 \\ 0,03 & 0,03 & 0,03 & 0,455 & 0,88 \\ 0,03 & 0,03 & 0,88 & 0,03 & 0,03 \\ 0,03 & 0,03 & 0,03 & 0,455 & 0,03 \end{bmatrix}$;

Ennek a mátrixnak egyértelmű sajátvektora van, $s=1$ sajátértékkel és ez az alábbi:

$$(0,43895; 0,43895; 0,51331; 0,51066; 0,28287).$$

Tehát a trükkel definiáltuk az egyértékű fontossági vektort.

3. Nézzük meg a teendőket a **nem-tiszta** hálózat esetén, vagyis amikor van olyan lap, amelyikre nincs hivatkozás, de ő használja a hálózatot. Miről is van szó? Az ilyen lapnak nincs fontossági száma, nem is tud „átadni” fontossági értéket az általa hivatkozott lapoknak, ezért azok fontossága alacsonyabb lesz, mint várnánk. A megoldáshoz *nincs elmélet*ünk, de számos programtechnikai trükk bevethető, amelyek valamilyen valószínűségi értékeken nyugszanak.

   Ilyen például, amikor az ilyen lapokat egy kezdeti fontossági értékkel látjuk el, számítva a rendszerbe való beépülésükre. Aztán a felülvizsgálati ciklus után elnyerik „sajátértékeiket”.

   Más a helyzet az olyan lapokkal, amelyek csak „leszívják a hálózatot”, de maguk nem tartalmaznak értéket, vagyis később sem lesz rájuk hivatkozás! Ezekhez tapasztalati értékeket rendelünk, hogy minden keresésük a találat lapok fontosságát növeljék! Például egy rendezés után a nulla fontossági értékű lapoknak egységnyi értéket adunk, amivel részt vesznek a hálózatban. Minél aktivabbak annál több fontosságot adnak át, tehát elértük a célunkat!

4. A *3w* **nem-monogám** hálózat, azaz akárhány kereső rendszer is működhet rajta! A valóság ennél sokkal korlátozottabb, hiszen érvényesül a járt utat járatlanért el ne hagyd elv. Nem könnyű új olyan kereső rendszert bevezetni, amely jobb eredményekkel kecsegtet, mint a már beváltak! Habár ki tudja, hiszen a mostaniakra is igaz volt ez egykor!

Egy biztos: nem törekedtünk abszolút „jó” megoldásra, talán nincs is, csak kerestünk egy „elegendően jó” megoldást. Mindenesetre sikeres disszertáció született!

Ebből lehetett milliárdos vagyonú céget építeni, hát szóval megérte! Nem?

A tudományos érdem a kitalálóké, a kidolgozóké, az őket támogatóké. A milliárdok a bevezetőké!

Így van ez jól a mai társadalmi rendszereinkben!?