A Matematikai Tudás Fája

Gilyén András Pál

Szakdolgozata

Témavezető: Dr. Lovász László
Számítógéptudományi tanszék

Eötvös Loránd Tudományegyetem
Természettudományi Kar

Tartalomjegyzék

1. Bevezetés

A szakdolgozat alapjául szolgáló ötlet az egyik egyetemi előadás alatt pattant ki a fejemből, amikor is az előadó kicsit elkalandozva a kiválasztási axióma igazságáról kezdett el filozofálgatni. Ekkor el kezdett foglalkoztatni az a gondolat, hogy mik azok a tételek és állítások amik ekkor nem maradnak igazak, vagy legalábbis nem tudjuk őket bizonyítani. Azt tudtam, hogy például a Banach-Tarski-paradoxon „megszűnne". De emlékeztem, hogy például analízisben is szóba került, néhány egészen alapvető állításnál is, az említett axióma. Vajon ismert-e rájuk a kiválasztási axiómát megkerülő bizonyítás, vagy akár alapvető analízisbeli állítások is elveszthetnék az érvényüket a kiválasztási axióma nélkül?

Ahogy ezen gondolkoztam a belső szemeim előtt megjelent egy nagy gráf, a „matematikai tudás fája", ahol alul vannak az axiómák, mint a fa gyökerei, és azoktól kezdve épülnek fel egymásra a különböző tételek és állítások, mint a fa ágai és leveli. És elképzeltem, ahogy egyszer csak a kiválasztási axiómát letöröljük, és ezzel sok-sok bizonyítás az érvényét veszti, és végül azok a tételek mind „elsárgulnak", amikhez csak a kiválasztási axiómát felhasználó bizonyítás volt ismert.

Később ez a kép a matematikai tudás fájáról nem hagyott nyugodni, és egyre pontosabban rajzolódott ki előttem. Mindig egyre több és több lehetőséget láttam benne! Bár tudtam, hogy hatalmas munka lehet összegyűjteni és rendszerezni, akár csak a matematikus Bsc képzés anyagát - nemhogy az összes ismert tételt. Mégis akkora lehetőséget láttam benne, hogy nem hagyott nyugodni a gondolat, hogy belevágjak próbaképpen a fa egy kis szegmensének a megalkotásába.

Amikor nekikezdtem a megvalósításnak, elsősorban a fa oktatási lehetőségeit tartottam szem előtt, mert az ilyen jellegű alkalmazásai, már egy kisebb részterület kidolgozásával is bemutathatóak. A célom egy működő prototípus [1] létrehozása volt, ami egy-két alapvető témakört dolgoz fel, és ezeken keresztül mutatja be az új forma képességeit. Végül is a választás az algebrára esett, mert itt gyakorlatilag előfeltételek nélkül lehet elkezdeni a fogalmak bevezetését. Részletesen a komplex számok bevezetését dolgoztam ki, és a lineáris algebra alapjait egészen a Jordan normálalak tételig.

A prototípus elkészítése során az elsődleges szempont az volt, hogy egy jól működő és használható programot írjak, de emellett végig szem előtt tartottam a későbbi bővíthetőség és tökéletesíthetőség lehetőségét is. Ezért egyből webes alkalmazásként kezdem el fejleszteni, a program nyelvének az objektum orientált Java-t választottam, és általában is a legmodernebb programozási technológiákat használtam fel. Arra is igyekeztem felkészíteni a programot, hogy a későbbiekben akár on-line is lehessen adatokat bevinni, ezáltal valamiféle wikipédia jellegű oldalt létrehozva.

2. A program

2.1. A modell

Először is fel kellett állítanom a matematika egy „modelljét”. Az volt az elképzelésem, hogy a matematika felfogható úgy, mint egy irányított hipergráf. A gráf csúcsai a definíciók és a tételek, az élek pedig a bizonyítások. A bizonyítás élek töve a bizonyításban felhasznált állítások, a feje(i) pedig a bizonyított tétel(ek). Ezen kívül a tételek illetve definíciók kimondásához szükséges fogalmak is alkotnak egy irányított hiperélet.

Ezután meg kellett alkotnom egy sémát, aminek segítségével a matematikai definíciókat, állításokat és azok bizonyításait egységes módon tudom eltárolni és kezelni. A definíció-tétel-bizonyítás hármas felosztás adta magát. Végül mégis úgy döntöttem, hogy a tételeket és a definíciókat alapvetően egységesen kezelem, hiszen a gráfban egyenrangú lesz a kétfajta fogalom, mindkettő csúcs lesz. Ráadásul a definíciók és a tételek matematikai értelemben sem különülnek el egymástól élesen, sokszor kontextus függő, hogy két ekvivalens definíció közül melyik a „valódi” definíció és melyik tétel. (Pl.: Egy $G \subseteq V$ által generált altér nem más mint a G-t tartalmazó összes altér metszete $\Longleftrightarrow$ Egy $G \subseteq V$ által generált altér nem más mint a $\lambda_1 g_1 + \lambda_2 g_2 + ... + \lambda_n g_n | \lambda_1, \lambda_2, ..., \lambda_n \in T; g_1, g_2, ..., g_n \in G$ halmaz.)

Így aztán egységesen kezelem és levélnek hívom a definíciókat, a tételeket és a példákat is, amelyek csupán egyetlen változó értékében különböznek egymástól. (Egy erre a célra bevezetett változó értéke mondja meg, hogy a levél éppen definíció, tétel vagy példa értelemben szerepel-e.) A levél elnevezés arra utal, hogy ezek lesznek a fa „levelei”, amelyek a számítógépes ábrázoláskor ténylegesen ki is fognak rajzolódni. Minden levélnek van egy címe, ami arra szolgál, hogy a felhasználó beazonosíthassa a mögöttes matematikai tartalmat. És végül, ami a lényeg, minden levélben el van tárolva, hogy melyik másik levelekre támaszkodik az általa reprezentál tétel/definíció kimondása. Vagyis a fogalmi függést kódolja le. Erről a fogalmi függésről feltettem, hogy tranzitív, ezért minden levél csak a legfelsőbb szintű levelekre támaszkodik. Például ha egy definíció használja a test és a vektortér fogalmát is, akkor direktben csak a vektortér fogalmától fog függeni, mert a vektorterek definícióján keresztül már úgy is függ a testek fogalmától. Ez fontos egyszerűsítés, ugyanis egy bonyolultabb definíció közvetetten rengeteg apró definícióra támaszkodhat, amit mind betáplálni sok fáradtságot jelentene, és kirajzoláskor is átláthatatlan ábrát eredményezne.

A bizonyítások sémája is hasonló, itt is van egy azonosításra szolgáló cím. Itt is van egy lista ami a felhasznált fogalmakat tárolja. És van egy lista, ami a bizonyított tétel(ek)re való hivatkozás(oka)t tárolja, valamint van még egy plusz változó is, ami a bizonyítás jellegéről mond valamit. Ez kódolja a bizonyítás típusát, ami lehet rendes bizonyítás, egyszerű következmény vagy speciális eset. A speciális eset tehát egy kicsit többet is kódol mint bizonyítást, implicite benne van egy állítás is: A speciális esete B-nek.

Ezek voltak az alapobjektumai a számítógépes megvalósításnak. Még egy fontos objektum típus van, aminek példányai az előbbiek rendszerezésére szolgálnak. Ezek a felépítések. Egy felépítés lényegében egy levelekből és bizonyításokból álló sorozat. A fogalmak olyan sorrendben jönnek elő, mint ahogy például egy egyetemi előadás során a táblára kerülnek a definíciók-tétetlek-bizonyítások. A jobb áttekinthetőség érdekében a felépítések egymásba is ágyazhatóak, vagyis a listában szerepelhetnek egész témakörökre való hivatkozások. Azt is el lehet dönteni, hogy egy hivatkozott felépítés a teljes listájával együtt kibontva kerüljön bele a tartalmazó listába, vagy csak közönséges levélként kerüljön be, úgy mintha csak egy összetett definíció/tétel lenne.

Még egy fontos kiegészítés tartozik a felépítésekben található listához, ugyanis a listában szereplő minden egyes levélhez/bizonyításhoz tartozik egy szám, ami azt hivatott jelezni, hogy az adott felépítésen belül mennyire számít fontosnak az adott lépés. Ezek a számok a kirajzolásnál, a nagyobb felépítések áttekintő ábráinak elkészítéséhez lesznek majd felhasználva. A 0 reprezentálja az elengedhetetlenül fontos leveleket/bizonyításokat. Minél nagyobb ez a szám annál inkább lényegtelen/technikai jellegű az állítás/bizonyítás. Ezek a lényegességi mérőszámok a részfelépítésekben minden szinten összeadódnak, és egyel növekednek. Vagyis ha egy A felépítés B részfelépítése az A listájában 5-ös lényegességi mérőszámmal rendelkezik, a B felépítés listájában egy C levél 2-es lényegességi szinten van, akkor az A felépítés felől nézve a C felépítés 8-as szinten van.

2.2. Ábrázolás

A megjelenítés alapegységei a felépítések. Minden felépítést ki lehet rajzoltatni az alkalmazással, a felépítéshez tartozó fát automatikusan legenerálja a program. A fának természetes számokkal indexelt emeletei vannak és minden levél pontosan egy emelethez tartozik. Ezeken a szabályokon belül a program 2 szempont szerint optimalizál, a következő prioritási sorrendben:

(i) Ha két levél közül az egyik a másikra támaszkodik, akkor a támaszkodó levél emeletének indexe legyen a nagyobb. Ha ez nem megoldható (azaz a gráfban van irányított kör), akkor a cél, hogy a felépítés sorrendje szerint a lehető legkésőbb következzen be ezen elv megsértése.

(ii) A fa legyen a lehető leglaposabb, azaz minden levél emeletének indexe legyen a lehető legkisebb.

Ezek az egyszerű szabályok egyértelműen meghatározzák, hogy egy levél melyik indexű emelthez fog tartozni. A sorrend meghatározásához egy egyszerű mohó algoritmust használhatunk. A mohó algoritmus egyszerűen addig telíti a gráfot a felépítés sorrendjében az élekkel, amíg egy új él nem zár kört. Ha kört zárna, akkor egyszerűen átugorja a listában és így folytatja a telítést. (Most gondolhatunk úgy a hiperélekre, mint egyszerű irányított élek halmazára, amiket csak egyszerre vehetünk hozzá a gráfhoz. A hipergráfban akkor van irányított kör, ha a hiperéleket az előbbi egyszerű élek halmazával lecserélve, a kapott közönséges digráfban van.) Így végül egy irányított kör nélküli gráfhoz jutunk, amiben tehát van topologikus sorrend. Most következik a második szempont szerint optimalizáló mohó algoritmus, ami egyszerűen topológikus sorrendben végigmegy a gráfon, és azokat a csúcsokat hozzáveszi az éppen aktuális (kezdetben 0-dik) szinthez, amibe nem futnak be élek, majd ha végig ért, az aktuálisan feltöltött szint csúcsait törli a segédgráfból, és a következő szintre ugrik. Ez az algoritmus $O(n^2)$-es, úgyhogy teljesen megfelel a célnak. Ha egy felépítésen belül más felépítések is vannak, akkor lényegében a listák uniójára alkalmazza a program a fenti eljárást.

Ezek után a kirajzolás kapcsán csak az a kérdés maradt, hogy egy adott szinten belül milyen sorrendben, és hova kell elhelyezni a leveleket. Elsődleges szempont, hogy a részfelépítések állításai egymás mellett/alatt legyenek, ez elengedhetetlen az ábra átláthatósága miatt.
A szinteken belül a sorrendet ezek után a program úgy határozza meg, hogy egyszerűen középről kezdi a levelek felhelyezését, és alternálva rakja jobbra balra a további leveleket attól függően, hogy hol van több hely. (Minden szint magassága egyforma, a levelek szélességét a címkéjük - azaz rövidített címük hossza szabja meg.)

A kirajzolás tehát úgy történik, hogy a fenti módszerrel kiszámolt szintekre kerülnek a levelek, a cím rövidítésével felcímkézve. A definíciók címkéjének háttere fekete, a tételeké zöld és a példáké barna. A fogalmi függéseknek megfelelő élek piros színűek, a közönséges bizonyítás élek színe kék, a következményé zöld és a speciális eseté narancssárga. A címkék hossza a hely jó kihasználása miatt általában nem haladja meg a 15 karaktert, és ezért sokszor elsőre nehezen értelmezhetőek, de ha az egeret a felhasználó egy címke fölé viszi, akkor a rendes címet is kiírja a képernyőre a program.

A tanulási és megértési folyamat elősegítése érdekében egy vízszintes csúszkát is elhelyeztem az alkalmazás tetejénél, amit léptetve haladhatunk előre a felépítés lépései szerint. Ekkor csak a listában már korábban szereplő levelek illetve bizonyítások jelennek meg az ábrán, és így lehet nyomon követni lépésről-lépésre, hogyan nő fel a fa. Ezen kívül a nagyobb anyagok fokozatos átlátásához nyújt segítséget a precíziós zoom, ami a dolgozat elején már ecsetelt fontosságot leíró változók alapján dönti el, hogy az adott szinten egy levél elég fontos-e a kirajzoláshoz. Így az adott felépítést egyre mélyebben és mélyebben szemlélhetjük, vagy ha már túl sok dolog van a képernyőn, amit nem látunk át, akkor „távolabbról" is ránézhetünk az ábrára. Ezen kívül helyet kapott még egy közönséges „optikai" zoom csúszka is baloldalt a precíziós zoom mellett.

A kirajzolást a fenti szabályrendszer összességében jól vezérli, a kapott ábrák eléggé jól követhetőek. De egy nyitott kérdés még maradt: a szinteken belül hogyan kell okosan elrendezni a levelek? A sorrend szabályozása fontos, mert megváltoztatásával egy szép és könnyen áttekinthető ábrából teljesen zűrzavarosat lehet létrehozni és fordítva. Ki kéne találni valamiféle olyan optimalizációs feladatot, ami a szinteken belüli elrendezéseknek az egész ábra áttekinthetőségére kifejtett hatását valahogy jól megfogja, és az optimalizálásra van is valamilyen polinomiális algoritmus. Kétfajta dolgot is találtam, amire lehetne optimalizálni, az egyik a minimális metszésszám, a másik az élhosszak négyzetének minimuma. Ezekre azonban csak exponenciális algoritmusok jutottam eszembe, ezért az idő rövidsége miatt egyelőre itt megálltam, mivel ez a kérdés most nem volt lényeges, és úgy gondolom, hogy ilyesfajta optimalizációs feladatokról önálló szakdolgozatot lehetne írni. Ideiglenes megoldásként, arra az esetre, ha a kialakult ábrában szerencsétlenül jönnének ki a szintek elrendezései, létrehoztam egy segédváltozót, amivel a felépítés szintjein beleüli levélsorrendet lehet módosítani.

2.3. Az enciklopédia

Az eddigiekben leírtam, hogy hogyan jeleníti meg a program grafikusan a tételek, definíciók és bizonyítások hálózatát. Ezek az áttekintő ábrák egy hagyományos könyv szemszögéből tekintve leginkább a tárgymutatónak vagy a tartalomjegyzéknek felelnek meg. A tartalom csoportosítására ebben a rendszerben ugyanis nem a fejezetek, hanem a felépítések szolgálnak. Ahogy egy hagyományos matematikai könyv legkisebb egységei a fejezetek pontjai, amik egy-egy állítást vagy bizonyítást részleteznek, úgy itt is a levelek illetve a bizonyítások a legkisebb (szöveg)egységek. A megjelenített fa egy levelére kattintva megjelenik annak részletes, matematikai igényességű leírása. A bizonyítások esetében úgyszintén, csak ilyenkor a bizonyított tétel címkéje felett megjelenő - az adott bizonyítást grafikusan reprezentáló félkörre kell kattintani. Illetve, ha egy részfelépítésre kattintunk, akkor az adott felépítés a program keretén belül egy új fülön nyílik meg.

Egy online enciklopédiától elvárható, hogy a szöveg a böngészőben jelenjen meg. De lévén matematikáról szó, legalább ilyen fontos, a képletek szép, áttekinthető megjelenítése is. Ezt a két követelményt egyszerre a legjobban a MathML [2] formátum teljesítette. A MathML 2.0 egy 2001-ben készült XML alapú szabvány, amely célja a matematikai szövegek webes beágyazásának egységesítése. Mára támogatja a Firefox webböngésző alapcsomagja és az Internet Explorerhez is letölthető ingyenes megjelenítő plugin. Viszont a MathML-nek az a baja, hogy az XML alapú szintaxisa miatt embertelen mennyiségűt kell gépelni egy egyszerű formula beviteléhez is. A formulákat tehát valahogy egyszerűbb módon kellene tárolni, hogy könnyen szerkeszthető legyen a tartalom. Ebből a szempontból leginkább a LaTex felelne meg. Némi keresgélés és próbálgatás után szerencsével jártam, és tudtam ötvözni a két megoldás előnyeit, ugyanis rátaláltam egy ingyenes Tex-to-MathML konverterre, a TtM-re [3]. Ez a konverter elfogadható szinten tudja kezelni a Tex kódot, és az átalakítás után kapott MathML kód kevés, kivételes LaTex szimbólumot leszámítva mindent képes visszaadni.

A véglegesen kialakult rendszer úgy néz ki, hogy minden matematikai alapobjektumhoz (levél ill. bizonyítás) tartozik egy Tex fájl a szerveren. Ha egy felhasználó meg akarja tekinteni például egy bizonyítás részletes leírását, akkor a szerver a TtM segítségével a Tex forráskódot átalakítja MathML-é, és az átalakított kódot küldi vissza a felhasználó webböngészőjébe. Szükség esetén a latex kódot is lekérhetjük a szerverről, és módosíthatjuk, ha valamivel nem vagyunk megelégedve.

2.4. Technikai részletek

A matematikai alapobjektumok adatait fájlrendszerben tárolom, minden objektumhoz egy külön XML fájl tartozik. A program belsejében az objektumok azonosítására a fájlrendszerben lévő elérési útvonaluk szolgál. Vagyis például egy felépítés létrehozásához elég ezeknek az elérési útvonalaknak egy tömbjét megadni, persze kiegészítve néhány segédváltozóval, amiről korábban már szó volt. Ez a rendszer azért kényelmes, mert eleve kizárja az azonosítók ütközésének lehetőségét. Továbbá az esetleges hibák lokalizálását is könnyebbé teszi, ugyanis az XML egy szöveges formátum, ami lehetővé teszi az adatok értelmezését és javítását, akár egy egyszerű szövegszerkesztő program segítségével is. Továbbá az XML formátum a könnyű bővíthetőség követelményének is megfelel, ugyanis egy új változó felvétele az adatstruktúrába nem teszi értelmezhetetlenné a korábbi, egyszerűbb struktúrával eltárolt adatokat.

A tudás fáját megjelenítő webes alkalmazás egy Java Applet. A szerverről egy Java Servlet Pages (JSP) program szolgálja ki az appletet, azaz küldi vissza a lekérdezett alapobjektumok leírásának MathML verzióját. Ez a konfiguráció szépen működik is a lokális szerveremen, de sajnos egyelőre még nem tudtam olyan komoly szerverre feltenni a programot, amelyiken elérhető a JSP támogatás. Ez pedig szükséges volna ahhoz, hogy a program összes funkciója on-line is működjön. Ameddig ez a szerver probléma nem rendeződik, átmeneti jelleggel, egy korábbi weboldalamon keresztül érhető el az alkalmazás. Itt csak PHP kiszolgáló van, emiatt az on-line szerkesztés nem üzemel, de egyébiránt a program tökéletesen működik.

A programot itt lehet megtekinteni: `http://mobilosiranytu.hu/matek.html`

3. Kitekintés

3.1. További lehetőségek

A program jelenlegi verziója jól bemutatja, hogy egy kisebb terület megismerése, illetve matematikai igényű felépítése során hogyan lehet a fogalmak rendszerezését segíteni és a témakörök átláthatóságát javítani, az összefüggések vizuális ábrázolása által. De ezzel nem merült ki a lehetőségek tárháza, sőt csak most kezd igazán látszani, hogy mennyi irányba lehetne továbbfejleszteni a módszert!

A legtöbb ötletem megvalósításához, illetve az új funkciók lehetőségeinek bemutatásához viszont jóval több, a rendszerben eltárolt matematikai ismeretanyagra volna szükség. Átfogó jellegű tudásanyag bevitele és rendszerezése azonban jócskán túlmutat egy szakdolgozat keretin. Egy embernek sokéves munkájába telhet, mire az alapoktól indulva komolyabb mélységekig felépíti, akár csak a legfontosabb matematikai témakörök palettáját is. Ezért van nagy jelentősége egy csoportos szerkesztési modell megalkotásának. Erre a lehetőségre már a kezdetek óta figyelmet fordítottam, az ilyen irányú kiterjesztés szempontjaira végig odafigyeltem, ahogyan ez a dolgozat korábbi részeiben látható is volt. Többek között ezért is kezdtem egyből on-line alkalmazásként fejleszteni a programot.

Az on-line szerkeszthetőség önmagában is sok kérdést vet fel. Szerkeszthesse-e bárki az oldalt - ezáltal lehetővé téve az oldal gyors gyarapodását - vagy csak egy szűk szakértő kör - ezáltal fenntartva az állandó színvonalat? Az általam elképzelt közösségi szerkesztési modell köztes megoldást választ a kétfajta előny ötvözése érdekében: Az oldalt bárki szabadon szerkeszthetné, mégis az ismeretlen szerkesztők által írt vagy módosított cikkek kezdetben a megbízhatatlan állományba kerülnének. Ezt követően a matematikailag képzett moderátorok, ha megfelelő színvonalúnak tartják, - eseteként a szükséges javítások után - beemelhetnék a megbízható tartalmak közé. A megbízhatatlan állomány tartalma ugyanúgy bárki számára megtekinthető lenne, mint a megbízhatóé, de a felhasználók dönthetnének úgy, hogy biztosra mennek, és csak a megbízható állományban nézelődnek.

Ha az on-line közösségi szerkesztői munka által, vagy esetleg valamilyen más formában már sikerült megoldani, hogy a programban tárolt adatok mennyisége és sokszínűsége elérjen egy jelentősebb szintet, akkor új dimenziók tárulkoznak fel. Például minden fogalom mellé el lehetne tárolni, hogy mikor keletkezett, és esetleg, hogy mikor alakult ki az alapjául szolgáló, de kezdetben még homályos elgondolás. Ennek segítségével jól lehetne ábrázolni a matematika fejlődését történeti szempontból. Lehetne látni, hogy mikor és hol nyíltak új területek, vagy fonódott össze két nagy matematikai irányzat. Ha a témakörök legalább egy része olyan, hogy a benne lévő fogalmak teljesen az alapokig vissza vannak vezetve, akkor lehetne vizsgálni az alapfogalmak és axiómák megkerülhetőségét is. Például meg lehetne valósítani a dolgozat elején felvázoltakhoz hasonlóan azt is, hogy vizuálisan elkülöníthető legyen, mely állítások maradnak érvényben a kiválasztási axióma nélkül, és melyek inognak meg. Az alapokig való visszavezetés tekintetében lehetne például vizsgálni a Bourbaki-csoport munkásságát, és összevetni más, kevésbé axiómatikus felépítésekkel. Vagy általában egy témakör különböző felépítéseinek analógiáit és különbségeit lehetne feltárni.

Az oktatás területén is vannak még nagy lehetőségek. Például bárki készíthetne saját felépítéseket a rendelkezésre álló tételekből, bizonyításokból. Vagyis egy oktató miután felment a honlapra, bejelölhetné az általa leadni kívánt anyag sarkalatos pontjait, és minden kijelölt tételre külön-külön kiválaszthatna egyet a lehetséges bizonyítások közül, például a legfrappánsabbat, a legegyszerűbbet vagy a legkevesebb előismeretet feltételezőt, annak megfelelően, hogy milyen jellegű kurzust szeretne tartani. És aztán, ha például a gyakorlaton egy diák valamilyen állításra találna egy frappánsabb bizonyítást mint ami az oktatott anyagban van, akkor azt utána feltölthetné a honlapra, amit utána a következő szemeszterben akár a többi oktató át is vehetne tőle.

Maradva az oktatás gondolatvilágánál, nagyon hasznosak lehetnének a példákat szemléltető ábrák, vagy akár a webes eszközöket felhasználó interaktív tartalmak. Például a Turing gép definíciója mellé be lehetne illeszteni egy egyszerű egyszalagos gépet megvalósító minialkalmazást, ami a lehetséges lépéseket szemléltetné, és esetleg még programot is lehetne rá írni. Vagy egy egyszerű animációval lehetne szemléltetni a Brouwer fixpont-tétel bizonyításában szereplő iteratív eljárást.

Még merészebben és messzebbre kalandozva, ha mondjuk már addig fejlődne a digitális tudástár, hogy a lényegesebb eredmények mind be lennének táplálva, akkor akár a jelenleg problémákkal küzdő matematikai folyóiratok szerepét is átvehetné az oldal. A legfrissebb cikkek egymásra épülve jelenhetnének meg az oldalon, és így minden kutató azonnal tudomást szerezne az ő témakörében megjelent új eredményekről.

Nem is folytatom tovább a lehetőségek felsorolását, mert félek, hogy egy idő után a matematikai témájúnak szánt szakdolgozatom közelebb kerülne a science fiction műfajához. Azt hiszem, hogy az eddig leírtakból is már látszik a lényeg: a határ a csillagos ég!

3.2. Összehasonlítás

Visszaereszkedve a földre, egy kicsit szemügyre szeretném venni, hogy milyen hasonló megoldások találhatóak ma az interneten.

Az on-line enciklopédiák területén vitathatatlanul a legsikeresebb példa a mára fogalommá vált Wikipédia [4]. A Wikipédia sikerét és gyors fejlődését, valamint azt a tényt is figyelembe véve, hogy komolyabb mennyiségű matematikai tartalom is található az oldalon, felmerülhet a kérdés, hogy van-e létjogosultsága egy újabb (matematikai témájú) enciklopédia kidolgozásának. A válasz véleményem szerint az, hogy igen, van. Az okok pedig a matematikai fogalmak rendszerének különleges szerkezetében rejlenek.

Míg egy átlagos szócikknél a többi szócikkel való kapcsolat megjelenítésére elegendő lehet néhány, a szövegbe beillesztett hivatkozás, addig a matematikai fogalmak precíz egymásra épülésének megjelenítésére ez elégtelen. Problémát jelent továbbá, hogy ugyanazon fogalom más és más felépítésekben egész másképp formalizálható. Egy filozófiailag kevésbé jelentős, technikai szempontból viszont fontos problémát jelent az is, hogy a matematikai kifejezések leírására legsűrűbben használt latex megjelenítésével kapcsolatban is vannak hiányosságai a Wikipédiának. A teljesen szabad szerkeszthetőség is problémás, nem lehet ugyanis állandó matematikai színvonalat fenntartani. Az általam elképzelt közösségi szerkesztési modell és az egyéb megoldások jobban illeszkednek a speciális tárgyhoz: a matematikához.

Az on-line matematikai tudástár iránti igény létezésének alátámasztására megemlíteném egy kis epizódot a szakdolgozat írásának történetéből: Az év elején az egyetemen találkoztam néhány volt oktatómmal az Operációkutatás tanszékről, épp miközben a programot fejlesztettem. Mivel érdeklődtek, hogy miből írom a szakdolgozatomat, hát gyorsan bemutattam nekik hogyan is néz ki az (akkor még félkész) alkalmazás. Aztán kiderült, hogy épp egy nagyobb feladatgyűjtemény rendbeszedését tervezik a tanszéken, és felmerült az on-line rendszerezés lehetősége is, de egy elérhető megoldás sem tűnt igazán a feladathoz illeszkedőnek. Viszont az én konstrukcióm nagyon tetszett nekik, és szóba került az is, hogy miután végeztem a dolgozattal, esetleg ki is próbálnák.

Az dolgozatban vázolt, és megvalósított elképzelésekhez talán leginkább hasonlító kezdeményezés a Sulinet Digitális Tudásbázis [5]. Ennek a Magyar Termék Nagydíjas oldalnak a célja a középiskolás tananyag online rendszerezése. Itt még az (elsősorban fizikai) fogalmak közötti összefüggéseket megjelenítő fogalomtérképek [6] is generálhatóak. Ám az összefüggések rendszerezése és osztályozása nem cél ezeken az ábrákon, inkább egyfajta asszociációs diagramként lehetne őket jellemezni, így jelentős koncepciós eltérés van az általam készített megoldáshoz képest.

Még valamelyest ide köthető egy, a világ matematikusainak adatait és kapcsolatai feltérképezni kívánó oldal [7] is. Csak itt a feltérképezés tárgya más, a matematika helyett annak művelői állnak a középpontban. Az internetes kutakodásaim során, az eddig felsorolt példák megoldásainál a dolgozatban vázoltakhoz lényegesen közelebb álló megvalósítással nem találkoztam.

Köszönetnyilvánítás

Végül szeretném megköszönni Lovász László tanárúrnak, hogy segített ennek a kicsit rendhagyó szakdolgozatnak az elkészítésben. Nagy biztonságot jelentett számomra ezen a kiforratlan területen, hogy mindig mutatta az utat, amelyen tovább kell haladnom. Köszönettel tartozom továbbá Pálfy Péter Pál tanárúrnak is, akinek a nagyszerűen felépített algebra előadásai alapján készítettem el a program matematikai tartalmát. És köszönöm családomnak és barátaimnak is, hogy bátorítottak és segítettek a dolgozat elkészülése közben.

Hivatkozások

[1] A Matematikai Tudás Fája, http://mobilosiranytu.hu/matek.html

[2] World Wide Web Consortium: Math Home, http://www.w3.org/Math/

[3] TtM, a TeX to MathML translator, http://hutchinson.belmont.ma.us/tth/mml/

[4] Wikipedia - The Free Encyclopedia, http://wikipedia.org

[5] Sulinet Digitális Tudásbázis, http://sdt.sulinet.hu

[6] Sulinet Digitális Tudásbázis: Fogalomtérképek, http://www.sulinet.hu/tart/cikk/Seb/0/32421

[7] The Mathematics Genealogy Project, http://genealogy.math.ndsu.nodak.edu