

Reaktorfizika alapjainak oktatása Python programkóddal

Elter Zsolt

Uppsala University, Division of Applied Nuclear Physics
Ångströmlaboratoriet, Lagerhyddsvägen 1, 751 20 Uppsala

Az Uppsalai Egyetem nyári kurzusaként kidolgoztuk a Reaktorfizika Python-nal című tantárgyat. Ennek különlegessége, hogy számítógépes laborgyakorlatokkal segíti elő az aktív tanulást. A gyakorlatok Jupyter Notebook-okra épülnek, melyek szöveges leírásokból, Python programkódból és vizualizációból állnak. Jelen cikk célja, hogy bemutassa a tantárgy nyilvánosan is elérhetővé tett tananyagát.

A tantárgy törzsanyaga lefedi a hasonló kurzusok által bemutatott témaköröket. A legtöbb gyakorlati feladatot az alapokból építettük fel, viszont az összetettebb számításokat a diákok az openMC kód felhasználásával végzik.

Bevezetés

Az Uppsalai Egyetemen több olyan tantárgy is elérhető a diákok számára, melyek által megismerkedhetnek a reaktorfizikával: egy BSc képzésben szereplő bevezető reaktorfizika kurzus, mely nem fizikusok képzését célozza (például a neutrontranszport-egyenletet nem tárgyalja); egy alkalmazott reaktorfizika tárgy, mely ugyan nem gyakorlati szempontból, de megismerteti a hallgatókat a reaktorfizikai számítások menetével; valamint egy *Energy Physics II with nuclear energy* (EPII) elnevezésű mesterkurzus, melynek az első fele a BSc-s tantárgynál komolyabb szinten tárgyalja a reaktorfizika elméletét, a második fele pedig a nukleáris fúzió elméletét fedi le. Ezen kurzusok során a diákok gyakorlati oktatása leginkább egyenletek levezetésével és analitikus megoldásával történt. Ezzel szemben a gyakorlatban a reaktorfizika jelentős programozási felkészültséget igényel, hiszen az egyenleteket numerikusan, vagy Monte-Carlo-módszerekkel oldják meg. Sajnos gyakori tapasztalatunk, hogy az egyetemen a fizikus hallgatók programozási felkészültsége viszonylag gyenge.

Ezért az elmúlt évek során az EPII kurzus beadandó feladataiként elkezdtünk *Jupyter Notebook*-okat készíteni, melyek segítségével a diákok egyszerűbb számítási és ábrázolási feladatokat végezhetnek Python könyvtárak használatával. Korábbi vizsgálatok azt mutatták, hogy az aktív tanulás növeli a hallgatók teljesítményét [1], és az ilyen feladatok bevezetése után a diákoktól mi is pozitív visszajelzéseket kaptunk. Ezért úgy döntöttünk, hogy kidolgozunk egy olyan önálló reaktorfizika tanfolyamot, amely még nagyobb hangsúlyt fektet a gyakorlati, aktív tanulási lehetőséget nyújtó feladatokra, és melynek célja, hogy a diákokat felkészítse haladó reaktorfizika kurzusok hallgatására. A tárgyhoz készült gyakorlati feladatokat *Jupyter Notebook*-ok segítségével, Python nyelven készítettük, mivel ezt a környezetet korábban sikeresen alkalmazták más tantárgyak kidolgozásánál is, mint például a Navier-Stokes egyenletek megoldásának oktatása során [2].

Jelen cikk célja, hogy röviden bemutassa a nyilvánosan elérhető gyakorlati anyagok tartalmát, ezáltal ötletet adva a reaktorfizikát oktató tanároknak. A cikk a NESTet 2021

konferenciára beküldött angol nyelvű cikk rövidített változata [3]. A 2. fejezet bemutatja a választott programnyelvet, és a programozási környezetet, valamint a tantárgy során használt openMC kódot; a 3. fejezet pedig összefoglalja a tantárgy anyagát.

Python, Jupyter Notebook, openMC és a számítógépes környezet

Ugyan különböző fórumokon állandó vita tárgyát képezi a programozó fizikusok között, hogy a Python megfelelő programnyelv-e tudományos programozási feladatok ellátására (általában a számítások lassúságát kiemelve, mint ellenérv), a tantárgy programnyelvének mi a Python-t választottuk. Ennek oka az, hogy a nyelv egyszerű szintaxisának és könnyű olvashatóságának köszönhetően a kezdő programozó számára a Python az egyik leggyorsabban elsajátítható programnyelv, ami előnyös egy olyan tantárgy esetén, melynek elsődleges célja nem a programozás megtanítása, hanem a reaktorfizika oktatása. Emellett, a gyakorlatok feladatai nem igényelnek jelentős számítási időt. Fontos kiemelni azt is, hogy a Python-hoz rengeteg tudományos könyvtár ingyenesen elérhető (pl. a *numpy* numerikus számításokhoz, a *scipy* differenciál-egyenletrendszer és numerikus integrálok megoldására, a *matplotlib* ábrázoláshoz, a *pandas* adatelemzéshez). A végső érvünk a döntés mellett az volt, hogy az openMC kódhoz elérhető egy kényelmesen használható Python API.

Továbbá úgy döntöttünk, hogy a tantárgyhoz a programozási környezet pedig a *Jupyter* lesz. A *Jupyter Notebook* egy olyan, böngészőben megjeleníthető interaktív dokumentum, amely kombinálja a szöveges dokumentációt a futtatható programkódot tartalmazó mezőkkel, valamint a dokumentumon belül megjeleníthető ábrákkal. A dokumentáció leírása a Markdown leíró nyelv segítségével történik, egyenleteket pedig LaTeX segítségével lehet létrehozni. Az 1. ábra az egyik ilyen gyakorlati anyag részletét mutatja. A diákok először elolvassák a szöveges útmutatást (a példában a neutron mozgási energiájának a csökkenését rugalmas szórás során), majd a feladatuk az, hogy a következő mezőben kiegészítsenek egy félkész

programkódot (a példában már a megoldás látható, a gyakorlati anyagban a 7-9 programsorok nem szerepelnek). Végezetül a diákok egy következő Markdown mezőben dokumentálják a következtetéseiket.

Mivel egy bevezető reaktorfizika kurzusban nem lehetséges (idő és kellő háttérismeret hiányában) összetettebb, heterogén reaktor-geometriákat vizsgálni, valamint bonyolult fizikai folyamatokat (például termikus szórás) bevezetni, ezért a valóságosabb esetek vizsgálatára (pl. nyomottvízes reaktor üzemyagcellája reflektív határfeltételekkel) az OpenMC kódot választottuk, mely egy, az MIT-n fejlesztett, Monte-Carlo-neutron- és fotontranszport-kód [4]. Ez a kód képes sugárforrásokat és sokszorozó rendszereket is kezelni. Az OpenMC kódot C++-ban írták, viszont elérhető hozzá egy olyan Python API, melynek segítségével a diákok előállíthatják az input fájlokat *Jupyter Notebook*-ban. A kód járulékos előnye, hogy ingyenesen elérhető, így a diákoknak nem kell felhasználói licenstértséget jelentkezni, valamint hogy a kód egyszerűbb geometriák esetén egy laptopon vagy személyi számítógépen is viszonylag gyorsan fut.

Mivel az openMC installálása Windows felhasználók számára körülményes, a tantárgyhoz készítettünk egy Virtual Box-szal futtatható virtuális Ubuntu környezetet,

melyben minden szükséges szoftvert és hatáskeresztmetszet fájlt előre telepítettünk.

A tananyag és a számítógépes laborgyakorlatok

A tantárgy kilenc előadása a bevezető reaktorfizika kurzusok megszokott témáit járja körül: magfizikai alapok (radioaktív bomlás, magreakciók, hatáskeresztmetszetek, maghasadás), neutronlassulás és -spektrum, neutrontranszport és -diffúzió, pontkinetika, mérések szubkritikus rendszerekben ($1/N$, Sjöstrand-módszer, Rossi-alfa) és kiégés. A szerző diákkori tapasztalatai alapján a hallgatók számára megtévesztő lehet, ha a tényleges fizikai tárgyalás és az egyenletek közelítő megoldási lehetőségei már egy bevezető tárgyban keverednek (pl. a neutrontranszport-egyenlet szemben a P_N közelítéssel). Így ez a tantárgy a determinisztikus reaktorfizikai számítási módszerekkel csak nagyon röviden, említés szintjén foglalkozik, valamint egy külön gyakorlat tartalmazza a diffúziós egyenlet numerikus megoldási lehetőségét. Ugyanakkor a törzsanyagban helyet kapott egy rövid bevezetés a Monte-Carlo-részecsketranszport-módszerekbe, mivel tapasztalatunk szerint ez sokkal jobban elősegíti a neutrontranszport intuitív megértését.

Experiment 1: Relation of angle and energy in elastic scattering

The neutron kinetic energy in LAB E'_l after elastic scattering depends on the energy before scattering E_l and the scattering angle θ_C in the CM

$$E'_l = \frac{1}{2} E_l [(1 + \alpha) + (1 - \alpha) \cos \theta_C]$$

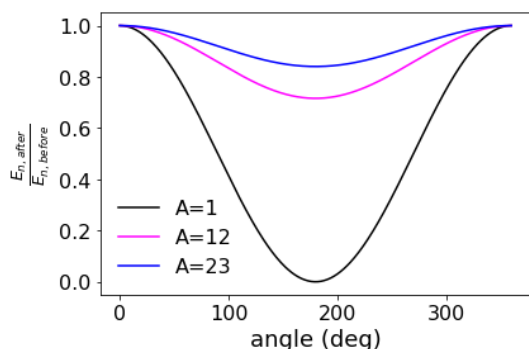
where

$$\alpha = \left(\frac{A - 1}{A + 1} \right)^2$$

and $A = M/m$

Let us investigate how the ratio of the incoming and the outgoing neutron energy depends on the scattering angle: Plot the above formula for several nuclides (eg. $A=1$, $A=12$, $A=23$, etc) and for angles between $0^\circ - 360^\circ$. Instead of repeating the same command, use a loop to iterate through all mass numbers.

```
In [1]: 1 import numpy as np
2 import matplotlib.pyplot as plt
3 theta=np.linspace(0,360,361)*np.pi/180
4 As=[1,12,23]
5 colors=['black','magenta','blue']
6 plt.figure()
7 for i,A in enumerate(As):
8     alpha=((A-1)/(A+1))**2
9     plt.plot(theta*180/np.pi,0.5*((1+alpha)+(1-alpha)*np.cos(theta)),colors[i],label='A='+str(A))
10 plt.xlabel('angle (deg)',fontsize=18)
11 plt.ylabel(r"$\frac{E_{n,after}}{E_{n,before}}$",fontsize=18)
12 plt.tick_params(axis='both', which='major', labels=16)
13 plt.legend(fontsize=16,loc=3,frameon=False)
14 plt.tight_layout()
15 plt.show()
```



1. ábra: Részlet egy Jupyter Notebook-ból

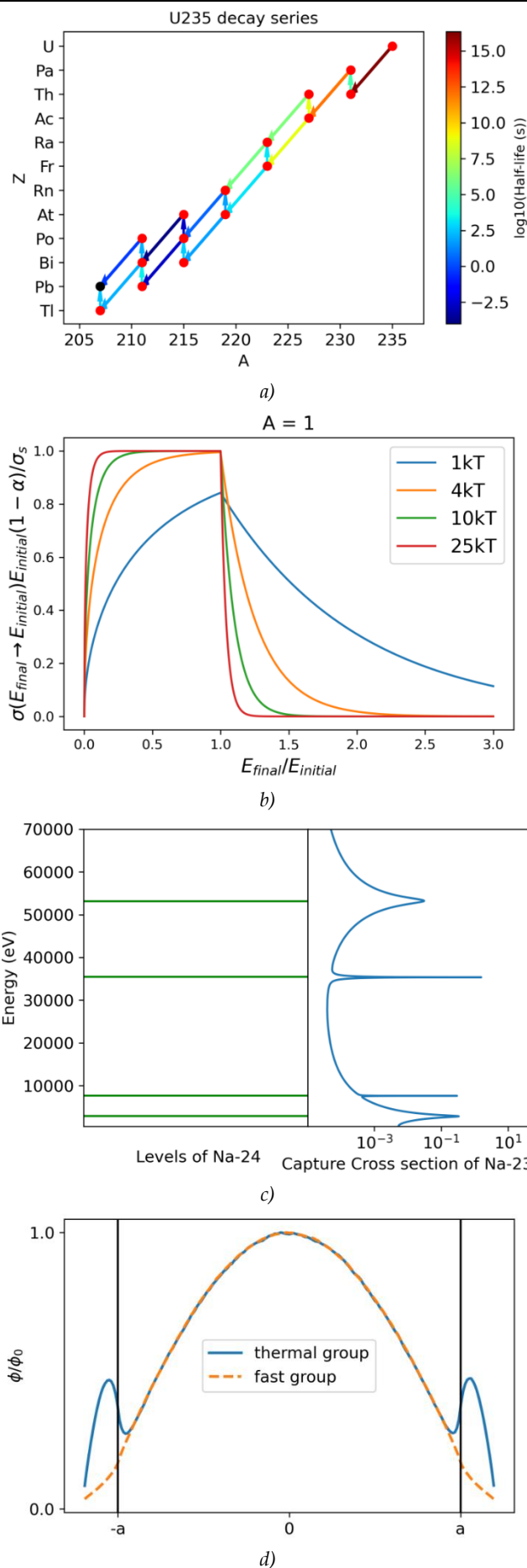
A tárgy célja az, hogy felkészítse a diákokat mind fizikai, mind programozási oldalról arra, hogy a jövőben komolyabb reaktorfizikai számításokkal foglalkozó tantárgyakat is elvégezhesenek.

Ennek a tantárgynak is ajánlott irodalma természetesen a Duderstadt és Hamilton által jegyzett *“Nuclear reactor analysis”* [5], de emellett a tananyagot egy ingyenesen elérhető óravázlatban is összefoglaltuk. Az óravázlat különlegessége, hogy a tudományos ábrákat nem más könyvekből kölcsönöztük, hanem Python-nal állítottuk elő, így a hozzátartozó programkód elérhetővé tételével a diákoknak lehetőségük van mélyebben is megérteni, hogy mi szerepel egy-egy ábrán. A 2. ábrán szereplő példák éppen ilyen tudományos ábrák. A 2. a) ábra az U-235 bomlási láncát, a 2. b) ábra a termikus szórási magfüggvényt mutatja. A 2. c) ábra a Na-24 mag energiaszintjeit veti össze a Na-23 magon való neutronbefogás hatáskeresztmetszetének rezonanciáival. A 2. d) ábra a termikus és gyors neutronfluxus térbeli eloszlását jeleníti meg egy reflektorral rendelkező reaktor esetén.

A tantárgy részét képezi 11 olyan számítógépes gyakorlat, melyek egy, esetenként két *Jupyter Notebook*-ot foglalnak magukba. A *Notebook*-ok úgy lettek megtervezve, hogy a diákok akár önállóan, tanári segítség nélkül is elvégezhetik a gyakorlatot, hiszen abban minden szükséges instrukció szerepel. Némely laborgyakorlathoz pár perces kiegészítő előadást is készítettünk (pl. hogy bemutassuk, hol lehet fellelni a szükséges hatáskeresztmetszet adatokat). Az első három gyakorlat a Python nyelv, és a gyakorlatok során később használt könyvtárak bemutatását szolgálja, míg a többi gyakorlat már magára a reaktorfizikára fókuszál. Minden laborgyakorlatban több feladat is szerepel, ezeket *“kísérletnek”* nevezzük, ezzel is nyomatékosítva, hogy a diákok szabadon változtathatják a különböző paramétereket, hogy a saját érdeklődésük szerint fedezhessék fel a fizikát. Ezen kísérletek vagy az 1. ábrán mutatott példához is hasonló hiányos programkódot tartalmaznak, amelyet a diákoknak kell kiegészíteni, vagy egy teljes programot, melynek a paramétereit változtatva vizsgálhatják, hogyan változik a megjelenített eredmény.

A továbbiakban röviden ismertetjük a 11 laborgyakorlatot. A feladatok egy részét a Duderstadt-Hamilton könyv, valamint a Szatmáry-féle reaktorfizikai feladatgyűjtemény [6] inspirálta:

1. Bevezető Python: Ugyan előkövetelmény, hogy a diákok alapvető programozási ismeretekkel rendelkezzenek, de a Python nyelv ismerete nem elvárás, ezért az első gyakorlat a Python nyelv szintaktikai szabályaira összpontosít, mely során a diákok megismerkednek az adattárolás formáival, ciklusokkal és elágazásokkal, függvény definíciókkal, és az alapvető I/O műveletekkel.
2. Haladó Python I: A gyakorlat során bemutatjuk a fontosabb Python könyvtárakat, a diákok folytonos hatáskeresztmetszet-adatokat és kétdimenziós függvényeket ábrázolnak, melyet a 3. a) ábra szemléltet olyan kétszer folytonosan deriválható függvény esetén, mely eltűnik egy szabályos hatszög peremén. Valamint a *scipy* használatával egyszerűbb bomlási láncok differenciális egyenletrendszerait oldják meg numerikusan.



2. ábra: Példaábrák az óravázlatból

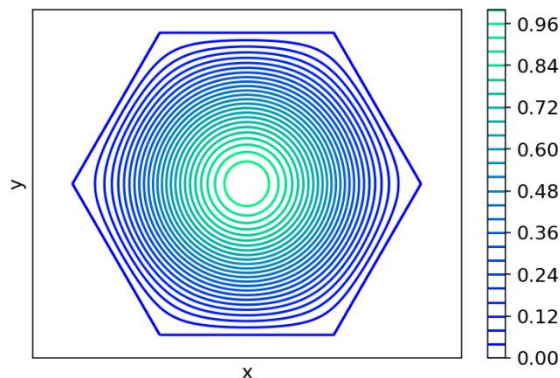
3. Haladó Python II: Mivel a későbbi gyakorlatok során a diákok openMC-vel dolgoznak, ezeken a laborokon elsajátítják az ehhez szükséges Python tudást, úgy mint az osztályok definícióját, valamint az adatelemzést a *pandas* könyvtárral. Ezen kívül bemutatjuk azokat az adatformátumokat, és azok olvasási módjait, mellyel a modern kódokat használó nukleáris mérnök később találkozhat (JSON, csv, xml).
4. Neutronszórás és Monte-Carlo-módszerek: A diákok rugalmas neutronszóródással kapcsolatos kísérleteket végeznek, például megvizsgálják a tömegközépponti rendszerben izotróp rugalmas szórás anizotrópiáját a laboratoriumi rendszerben, valamint a neutronok termikus energiákra való lassításához szükséges ütközések számát a tömegszám függvényében. A gyakorlat során megismerkednek a Monte-Carlo-módszerek alapjaival, diszkrét és folytonos eloszlásfüggvények mintavételezésével, például az ütközések közti véletlen úthossz mintavételezésével, vagy a hasadásban keletkező neutronok energiájának sorsolásával, ahogy ezt a 3. b) ábra szemlélteti, melyen éppen a Watt-spektrumot mintavételezzük Neumann-féle rejekciós módszerrel.
5. Véletlen hasadási fák és Monte-Carlo-transzportszámítások: A gyakorlat első felében a diákok egy egyszerű programmal különböző hasadási fákat generálnak a hasadás valószínűségét variálva. A 3. c) ábra ilyen véletlen fát illusztrál a neutrongenerációk mentén (piros kereszt jelzi azon neutronokat, melyek nem vesznek részt hasadásban, míg fekete pont jelzi azokat a neutronokat, melyek hasítanak, és a hasadásban a felhasználó által beállított eloszlás szerint új neutronok születnek). Majd numerikus optimalizációs módszerrel megállapítják, hogy milyen valószínűség mellett lesz kritikus a rendszer. Az így kapott eredményt összevetik az elméleti várakozásokkal. A gyakorlat második felében egy egyszerű (homogén, 0 K hőmérsékletű hidrogénatomok közé helyezett pontforrás) sztochasztikus részecsketranszport vizsgálatát végzik. Az ehhez szükséges program tulajdonképpen a 4. gyakorlat során előállított mintavételező függvények összecsatolását hajtja végre. A hidrogén sűrűségét változtatva vizsgálhatják a fluxus térbeli eloszlásának változását.
6. OpenMC bevezető: A diákok egy nyomottvízes reaktor üzemanyagcellájának a modelljén keresztül megismerkednek az összetettebb geometria leírásának módjával. Kiszámítják a végtelen sokszorozási tényezőt, a neutronok spektrumát, valamint a termikus és gyors fluxus térbeli eloszlását. Megvizsgálják, a rezonancia-önárnyékolás hatását, valamint azt, hogy a termikus fluxus a pácán belül alacsonyabb, a gyors fluxus pedig nagyobb, mint a pácá körüli moderátorban.
7. Egycsoport neutrondiffúziós egyenlet megoldása egydimenziós esetben numerikus módszerekkel: a gyakorlat elején a diákok megtanulják, hogyan lehet térben diszkretizálni, és mátrixegyenletként felírni a diffúziós egyenletet, majd megoldják nem-sokszorozó közegben időben állandó forrás mellett, és sokszorozó közegben is. A gyakorlat célja, hogy a diákok megismerkedjenek a determinisztikus reaktorfizikai számítási módszerek alapjaival. A 3. d) ábra azt szemlélteti, ahogyan a diákok megfigyelhetik, hogy a fluxus térbeli eloszlása sokszorozó közegben a megoldáshoz konvergál a hatványmódszer alkalmazása során. A gyakorlat végén folytonos hatáskeresztmetszeteket súlyoznak egy előre kiszámolt neutronspektrummal azért, hogy megismerkedjenek a csoportállandók kiszámításának alapjaival. Az érdeklődő diákok azt is megtanulhatják, hogyan lehet az openMC kódot csoportállandó-számításra használni.
8. Reaktorkinetika: A gyakorlat során lépésenként oldjuk meg az időfüggő kinetikai egyenletet. Először csak prompt neutronokat vesszünk figyelembe, majd egy átlagos későneutron-anyamag-csoportot, végül eljutunk mind a hat későneutron-anyamag-csoportot figyelembe vevő esethez. A gyakorlat végén ábrázoljuk a reciprokóra-egyenletet, és adott reaktivitásbevitel mellett meg is keressük az egyenlet gyökeit.
9. Szubkritikus rendszerek: Ez a gyakorlat ízelítőt ad a kísérleti reaktorfizikából. A diákok mért adatokat elemeznek. Az $1/N$ módszer segítségével neutrondetektorok beütés-száma alapján megbecsülik, hogy mekkora reaktivitás juttatható be a reaktorba anélkül, hogy az szuperkritikussá válna. Pulzált neutronforrásos mérés során rögzített detektorjelekből, Sjöstrand-módszer alkalmazásával megbecsülik a szubkritikus reaktivitást. Végezetül egy korrelált beütéseket szimuláló program használatával Rossi-alfa méréseket végeznek.
10. Haladó openMC: A gyakorlat során a diákok openMC számításokat automatizálnak. Ezáltal megállapítják a korábban vizsgált végtelen üzemanyagcella üregegyütthatóját, valamint tanulmányozzák, hogyan változik a végtelen sokszorozási tényező (k_{∞}), a rezonanciakikerülés valószínűsége (p), és a termikus hasznosítási tényező (f) az üzemanyag-moderátor arány függvényében. A gyakorlat megoldását a 3. e) ábra mutatja. Értékes tapasztalatokat gyűjtenek alul- és felülmoderált reaktorokról.
11. Kiegészi számítások openMC-vel: A diákok kiegészi számításokat végeznek openMC-vel, és megvizsgálják néhány nuklid koncentrációjának a változását a kiegészi függvényében, valamint a neutronspektrum változását besugárzás során, ahogy ezt a 3. f) ábra szemlélteti. A gyakorlat egyik fontos üzenete, hogy a Bateman-egyenletek megoldása a Monte-Carlo-alapú kódokban is determinisztikusan történik, a szerző tapasztalata szerint ezt sok kezdő kódfelhasználó félreírta.

Fontos kiemelni, hogy a gyakorlatok célja az volt, hogy a részletes leírásokkal és a félig megírt programkódokkal segítsük és ösztönözzük a diákokat az önálló munkára. A szerző személyes tapasztalata, hogy ez sokkal hatékonyabb módja a tanulásnak, mint nagyszámú beadandó feladatot kiadni tényleges útmutatás nélkül. Hasonló következtetésre jutott a [7] szerzője is, miután aktív tanulási elemeket vezetett be egy hasonló reaktorfizika kurzusban. Érdemes megjegyezni azt is, hogy a laborgyakorlatok elengedhetetlen része volt, hogy részletes visszajelzést küldjünk a diákoknak, rámutassunk, ha valamit elrontottak a megoldás során, vagy ha a konklúziójukból kimaradt valami fontos észrevétel (például az 5. gyakorlat során nem minden diák veszi észre, hogy a neutronfák esetén a kritikussághoz tartozó hasadási valószínűség a hasadásban

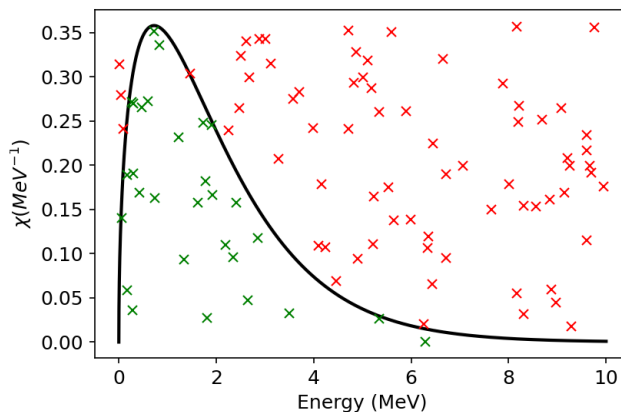
keletkező neutronok várható értékének a reciproka, ezért ezt fontos nyomatékosítani).

A laborgyakorlat mellett a számonkérés összesen nyolc beadandó feladattal történik. Ezekhez a feladatokhoz a diákok olyan *Jupyter Notebook*-okat kapnak, melyek csak a probléma leírását és a szükséges változókat tartalmazzák. A beadandók a gyakorlati anyagokra épülnek (pl. egyik feladat az, hogy készítsenek egy olyan Monte-Carlo-szimulációt, amivel meghatározzák egy U-235 gömb kritikus sugarát, melyhez a 4. és 5. gyakorlatokban készített függvényeket kell használni). Itt a diákoknak már nagyobb szabadságuk van a megoldást és az alkalmazandó kódot illetően.

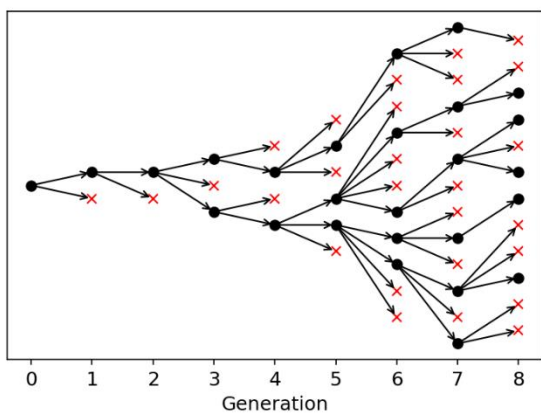
Mivel a gyakorlatok és a beadandó feladatok már alaposan felméri a diákok tudását, ezért a végső számonkérő vizsga helyett a diákok haladó reaktorfizikai témakörökből (pl. neutronzaj, reaktordinamika, transzportegyenlet megoldási módszerei, Shannon-entrópia alkalmazása kritikusági számolásokban) választhatnak, majd – az általunk kiosztott, kapcsolódó cikkek és könyvfejezetek feldolgozásával – egy 2-3 oldalas esszében, valamint egy 5-10 perces, videó-előadásban számolnak be a választott témáról. Ezek az előadások elérhetőek a többi diák számára is, így a diákok maguk bővítik ki a tantárgy anyagát.



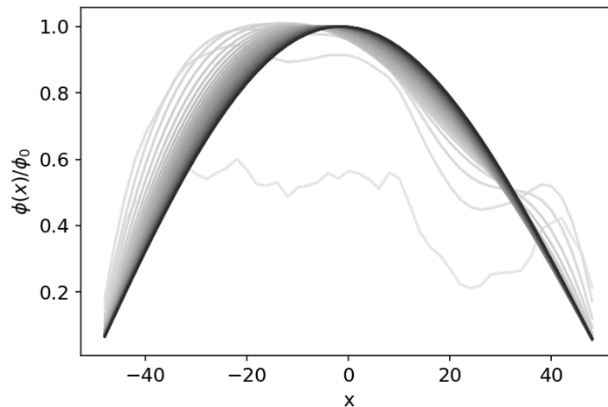
a)



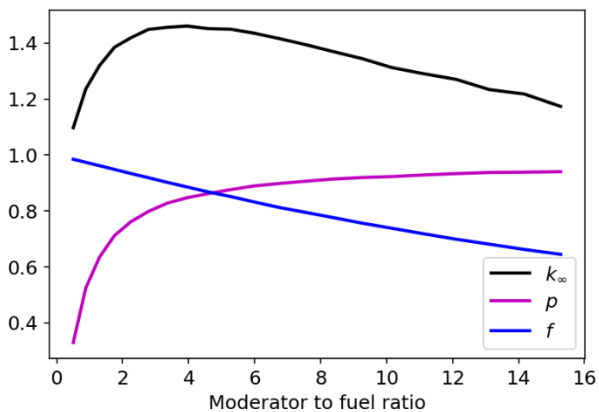
b)



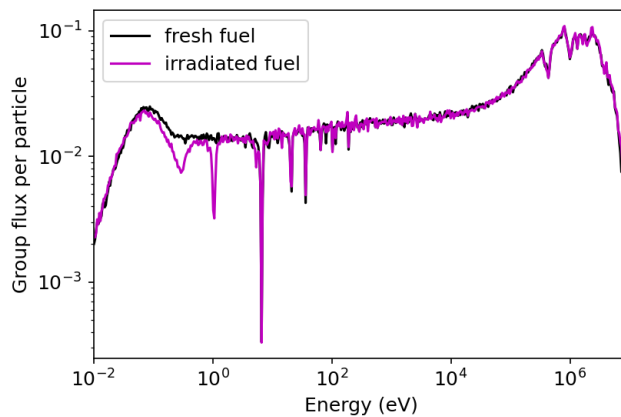
c)



d)



e)



f)

3. ábra: Példaábrák a gyakorlatokról

Összefoglalás és jövőbeli tervek

A cikk egy új, aktív tanuláson alapuló reaktorfizika tantárgyat, főként annak számítógépes laborgyakorlatait mutatta be. A gyakorlatok *Jupyter Notebook* környezetben, Python nyelven írt programok, és az openMC neutron- és fotontranszport-kód segítségével ismertetik meg a diákokat a reaktorfizika szépségeivel. A gyakorlatok anyagai ingyenesen hozzáférhetőek és szabadon felhasználhatóak tanulás és tanítás céljára [8].

Ezt a tantárgyat eddig csak 2021 nyarán oktattuk, viszont az azt elvégző tucatnyi diák visszajelzései alapján sikeres volt (a hasonló uppsalái kurzusokhoz képest magasabb diákszám annak volt köszönhető, hogy a távtanulási lehetőség miatt a kurzust 4-5 svéd egyetem hallgatói is követték). A diákok a kurzus utáni értékeléskor úgy

nyilatkoztak, hogy a tantárgy ugyan sok erőfeszítést igényelt, de rendkívül élvezetes volt. Azt tapasztaltuk, hogy az aktív tanulási lehetőség biztosításával sokkal könnyebben rá lehet venni a diákokat, hogy elmélyüljenek a tananyagban, és azt sokkal hatékonyabban és mélyebben sajátítják el. A tudományos programozás lehetősége pedig olyan diákokat is becsábított, akik amúgy nem gondolkodnak nukleáris technikával kapcsolatos karrierben.

Mivel jelenleg az Uppsalai Egyetemen a diákoknak nincsen lehetőségük haladó reaktorfizika tárgyakat hallgatni, ezért tervben van, hogy folytatásként egy haladó, reaktorfizikai számításokat lefedő kurzust is fejlesztünk, melyben a hangsúly – az itt tárgyalt kurzushoz hasonlóan – az aktív tanulási lehetőségen és a módszerek alkalmazásán lesz.

Köszönetnyilvánítás

Köszönet, Andreas Solders-nek, akivel együtt kezdtük el bevezetni a Python alapú beadandókat az EPII kurzushoz. Hálás vagyok Henrik Sjöstrand-nak, hogy felkért a Reaktorfizika Python-nal című tantárgy kidolgozására a *Energy Systems Engineering* mesterképzés diákjai számára. Továbbá köszönet Markus Preston-nak, hogy magára vállalta a pontkinetikai feladatok és az óravázlat vonatkozó részének kidolgozását, valamint Erik Branger-nek, hogy segédkezett a szubkritikus rendszerekhez kapcsolódó gyakorlatok és jegyzetek előállításában.

Irodalomjegyzék

- [1] Freeman, S., et al., "Active learning increases student performance in science, engineering, and mathematics", *PNAS* 111(23), pp 8410. (2014) DOI: 10.1073/pnas.1319030111
- [2] Barba, L. A. Forsyth, G. F., "CFD Python: the 12 steps to Navier-Stokes equations", *The Journal of Open Source Education*, 1(9), 21, (2018) DOI: 10.21105/jose.00021
- [3] Elter, Zs. et al., "Reactor Physics with Python: An Active Learning Experience At Uppsala University", *NESTet 2021 konferencia* (2021)
- [4] Romano, P. K. et al., "OpenMC: A State-of-the-Art Monte Carlo Code for Research and Development," *Ann. Nucl. Energy*, 82, 90–97 (2015) DOI: 10.1016/j.anucene.2014.07.048
- [5] Duderstadt, J. J., Hamilton, L. J., "Nuclear Reactor Analysis", Wiley, (1976), <https://hdl.handle.net/2027.42/89079>
- [6] Szatmáry, Z., "Reaktorfizikai feladatok", Egyetemi jegyzet, Budapesti Műszaki és Gazdaságtudományi Egyetem (2007)
- [7] Demazière, C., "Using active learning in hybrid learning environments" *International Conference on Physics of Reactors: Transition to a Scalable Nuclear Future*, *PHYSOR 2020*, DOI: 10.1051/epjconf/202124714001
- [8] Elter, Zs. et al., "Reactor Physics with Python (2021 edition)" archive, DOI: 10.5281/zenodo.5578876