

Jože Zobec, 28031255

Računalniška orodja v fiziki

Projekt 4

Grafi II

Predgovor

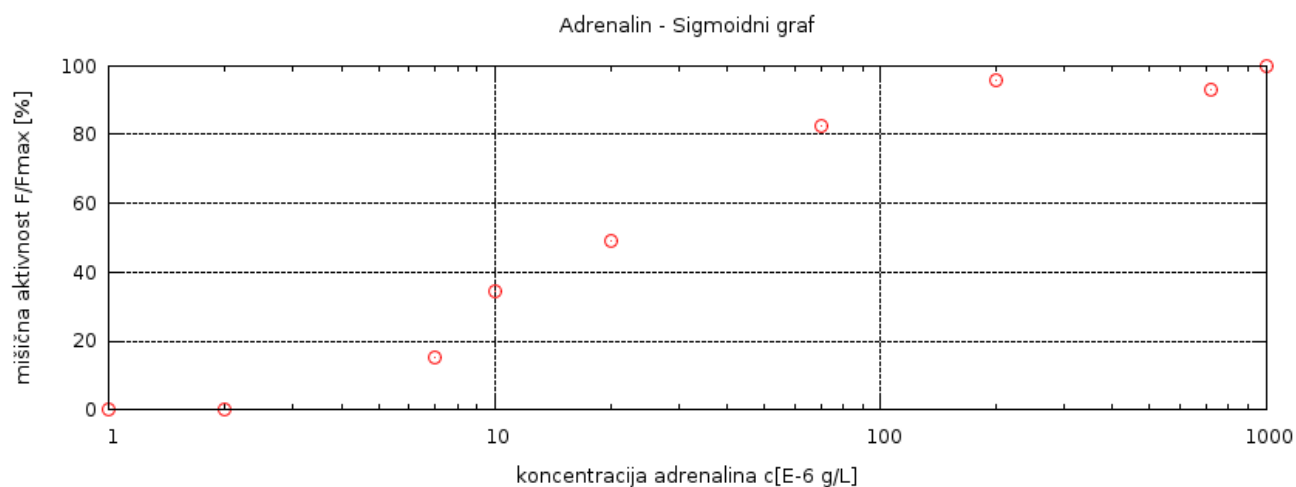
Pri reševanju sem si pomagal s programom GNUplot, ki lahko zelo elegantno dela z datotekami, riše po več funkcij, dopolni nekaj točk s poljubno krivuljo (za razliko od Excela), razume # simbole za komentarje (ki pa jih lahko zamenjamo s % ali kakšnim drugim simbolom; za komentarje si lahko izberemo tudi več simbolov, zavoljo kompatibilnosti z drugimi programi, npr. Matlab). Program je torej uporaben za obdelavo podatkov (datotek), ki vsebujejo glavo.

Zaradi preprostosti sem si ustvaril datoteke formata “.p”, ki sem jih nato minimalno dopolnjeval, v programu pa izrisoval z naslednjim ukazom: load “ime_datoteke.p”. Tako sem dobil izgled grafa in ko mi je bil dovolj všeč, sem ga shranil. Zadnjo nalogo sem rešil v programu Origin, ker bi sicer moral v gnuplotu ročno določiti in oblikovati vseh 28 oken, kar pa je nepotrebno s programi, ki jih imam pri roki.

Graf 1 – Adrenalin.dat

Graf res nekoliko spominja na črko S. V logaritemski skali bi moral še najbolj spominjati na $a1 \cdot \arctg(x-b1)+c1$ v linearni skali, vendar tu ni logaritemске skale (v slednji je $\arctg(x)$ veliko bolj strm in ni več tako ležeren, kakor v linearni).

Graf izgleda tako:



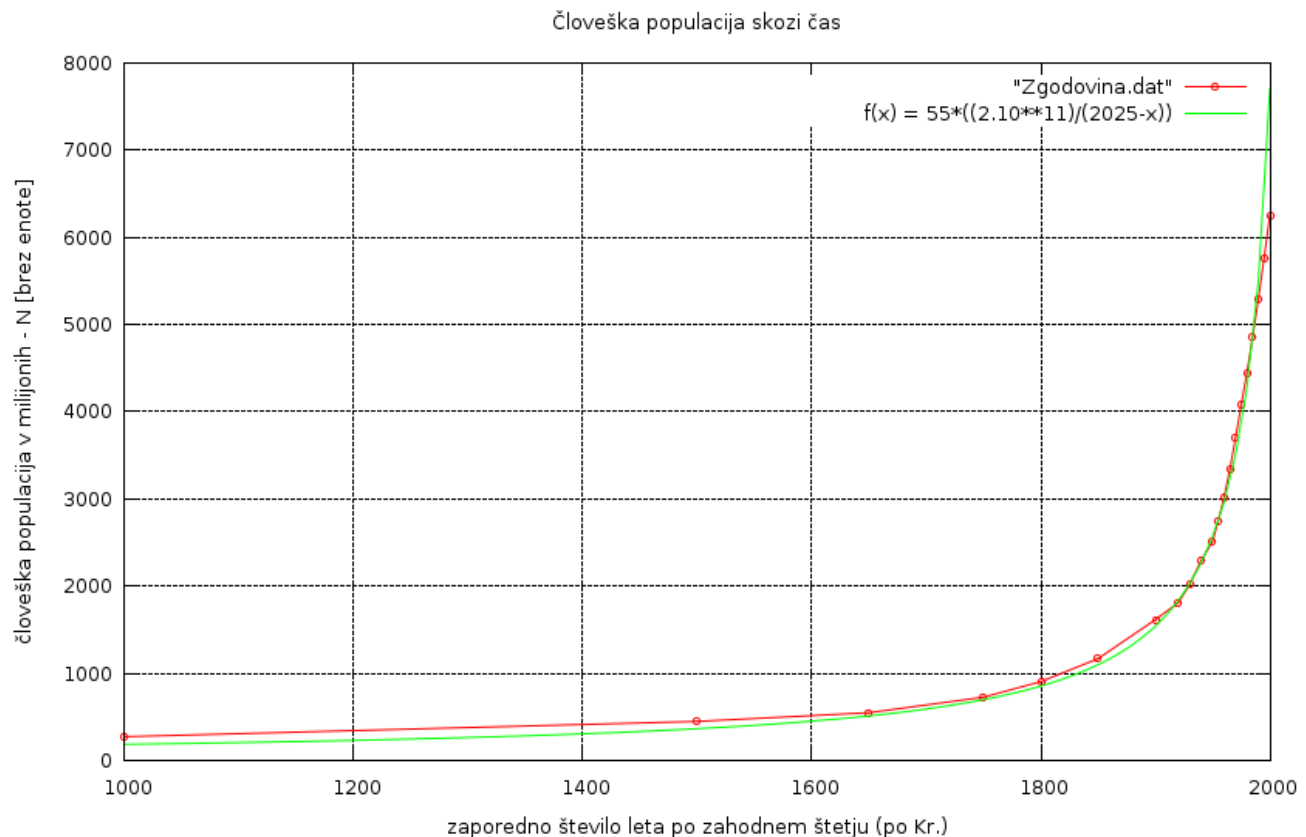
Graf 1.1 Graf res spominja na tako imenovano “sigmoido”, kot jo je na tablo narisal prof. dr. Kodre. Če bi simbole za točke povezali s črtami, bi graf bolj spominjal na premico – v meritvi so odstopanja – borih 10 točk pač ni meritev, ki bi bila dovolj zanesljiva, četudi le za povezavo točk s črtami (nismo gotovi, da so točke sploh tam).

Graf 2 – Zgodovina.dat (Kapicov model)

Za ta graf sem si zadal cilj, da bom narisal tako graf iz podatkov in ga nato primerjal z grafom funkcije $f(t) = 2 \cdot 10^{11} / (2025 - t)$.

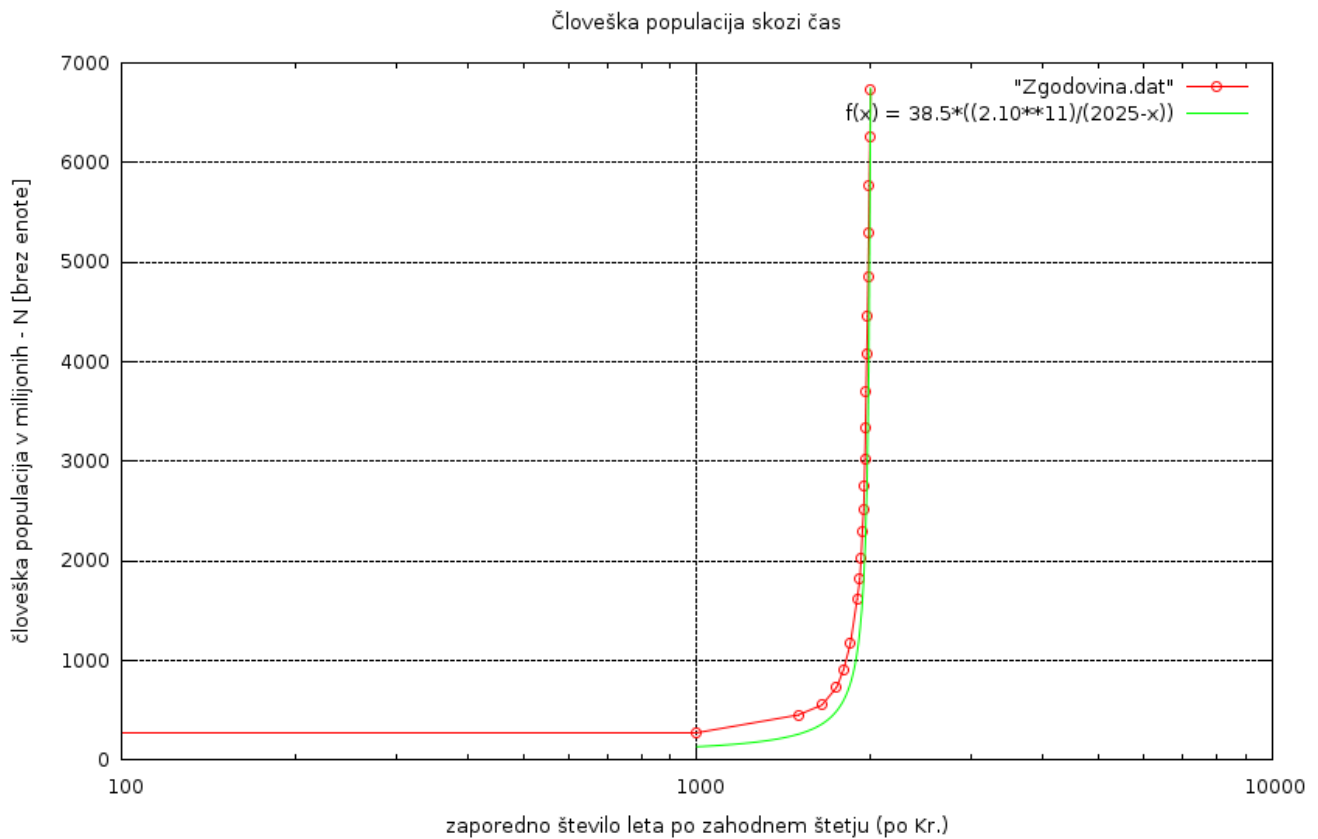
Graf je brez logaritemske skale skrajno neprimeren, saj tako po merjenih podatkih, kot po funkciji dobimo na videz premico (če zajamemo vse defenicijsko območje). Če na hitro odmislimo počasno ležerno naraščanje krivulje in se omejimo le na del grafa, kjer se dejansko začne kaj dogajati – na skok v populaciji – ocenil sem ga od leta 1000 do 2000. funkcijo $f(t)$ sem pomnožil s faktorjem

$$55 \cdot f(t) = f(x).$$



Graf 2.1 Zaradi nazornosti sem točke iz datoteke povezal s čro, za primerjavo dejanskih točk s krivuljo $f(x)$ sem dodal tudi točke. Kot vidimo, tako poveča krivulja kaže odstopanja, ki pa niti niso tako huda. Tudi graf, zožan na tako defenicijsko območje je kar zadovoljiv, vendar pa ne zadosti svojemu naslovu: "Človeška populacija skozi čas", marveč bi moral biti "Človeška populacija preteklih 1000 let". Graf $f(x)$ prične precej strmo naraščati, kar pa pomeni, da, pomnožen s tako velikim faktorjem ne bo več dober.

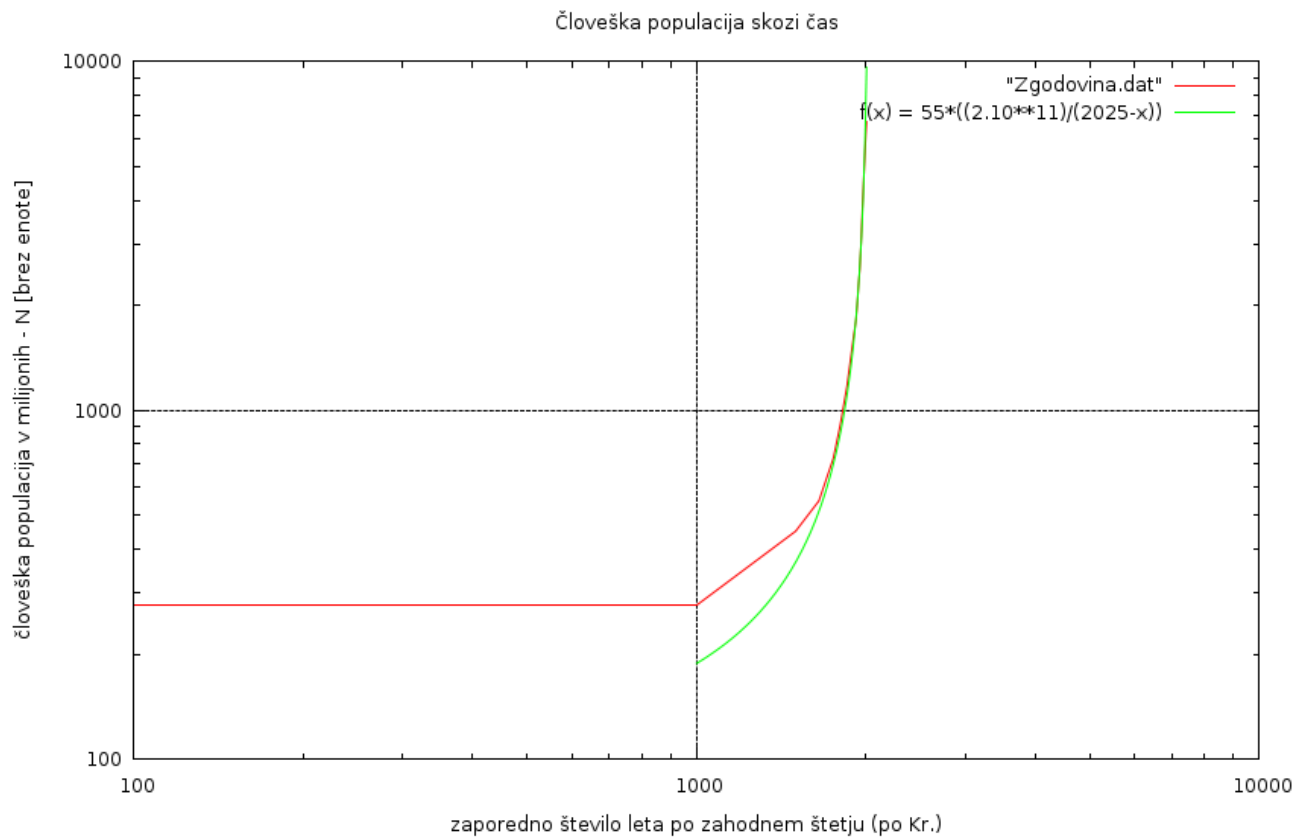
Zato, ker nas zanima celotno, oz. Čim širše defenicijsko območje, je treba dodati logaritemsko skalo na x osi:



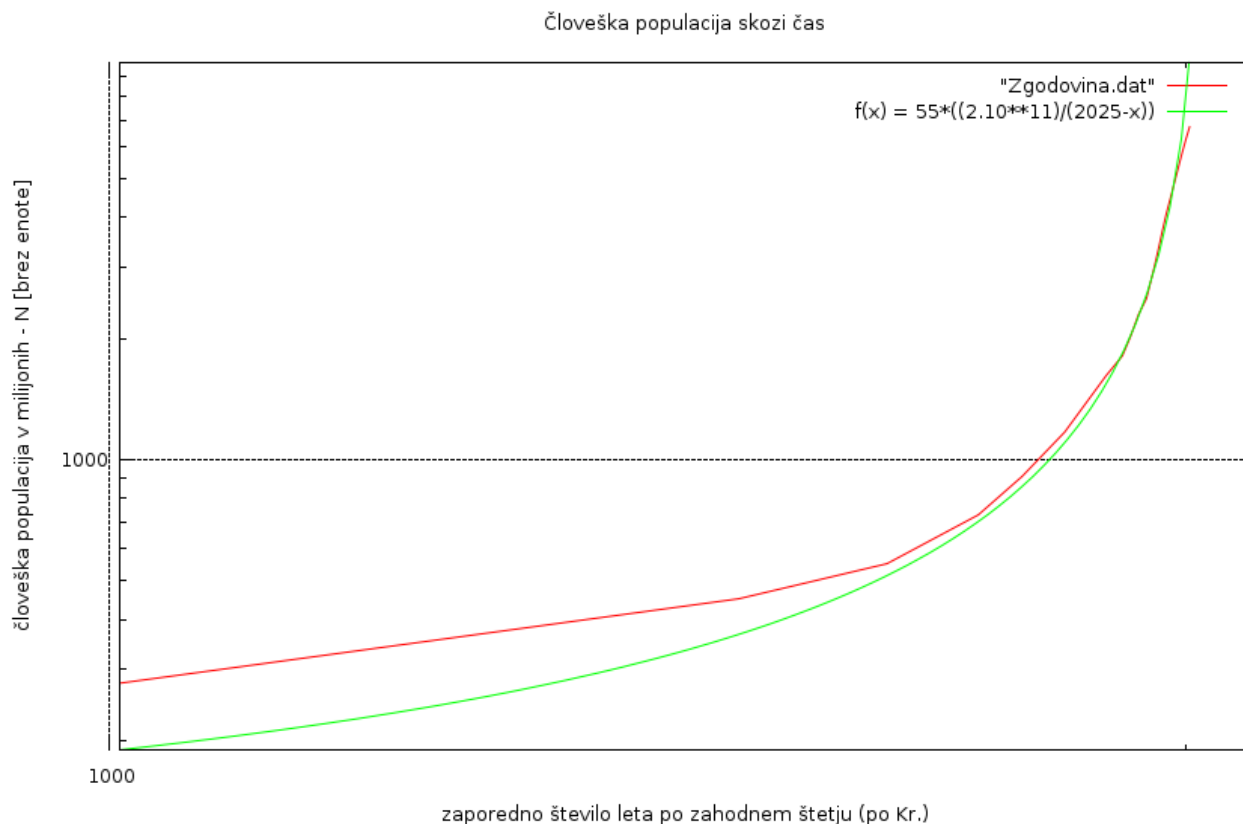
Graf 2.2 Da je boljše zajeto definicijsko območje sem dodal logaritemsko skalo na abscisno os. Funkcijo je bilo treba pomnožiti z nižjim faktorjem, zaradi česar je posledično večje neujemanje nižje pri dnu ordinatne, je pa zaradi tega boljše ujemanje pri vrhu (obe funkciji se končata na približno istem mestu).

Funkcija se tako na videz ujema s podatki iz datoteke, vendar tega ne vemo zagotovo, ker je skok velik (zelo strma funkcija). To nam da misliti, da bo treba poskrbeti z logaritemsko skalo opremiti tudi ordinatno os:

Funkcija $f(t)$, definirana kakor je, pomnožena z različnimi faktorji, se nič več tako dobro ne ujema (majhne razlike na logaritemski skali zelo narastejo).



Graf 2.3 Zaradi boljšega ujemanja na skoku sem spet pomnožil s faktorjem 55. Veliko neujemanje spodaj se pozna tudi pri nižjih faktorjih. Graf je še vedno strm. Ne zdi se mi primeren – že ena logaritemska skala je težka za branje, na dveh oseh je pa malo preveč. Zaradi prevelike gostote točk, sem jih umaknil točke povezal le s črto.



Graf 2.4 Prikazuje relativno dobro ujemanje funkcije $f(x)$ na skoku. Namen tega grafa NI matematično-fizikalni ampak je tu zgolj in samo zgolj za ponazoritev, zato je tudi bolj šlampast (skala na x osi), ker je v bistvu čisto nepotreben.

Najboljši izbiri se mi zdita grafa 2.1 (če nas zanima zelo ozko področje) in pa graf 2.2 (lepo prikaže naraščanje na sicer omejenem prostoru, vendar na celotnem definicijskem področju). Ker imamo omejen prostor, sem se torej odločil za **graf 2.2**, torej graf, ki ima le absciso v logaritemski skali (graf log-log se mi zdi preveč nepregleden).

Opomba: Zavaljo morebitne nedoslednosti sem poskušal narisati tudi funkcijo,

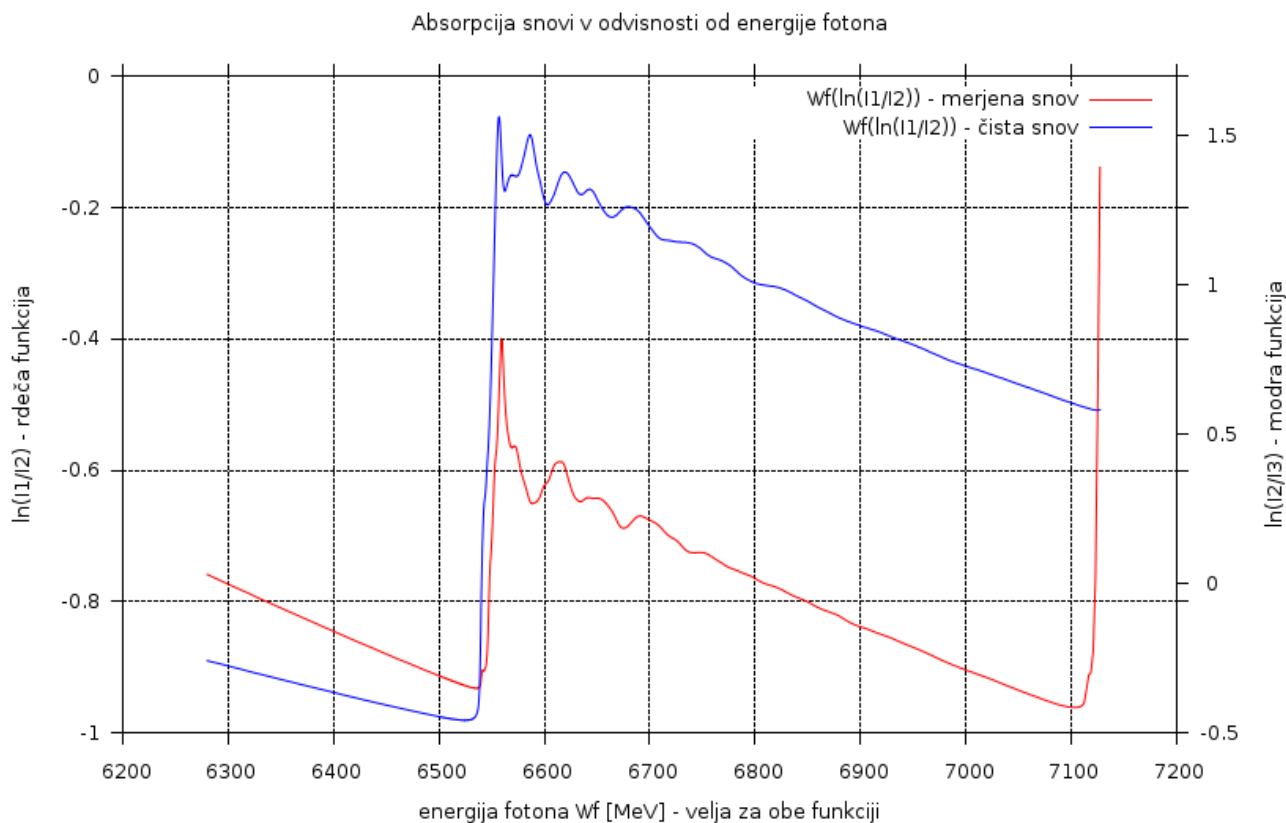
$$f(t) = (2 * (10^{11})) / (2025 - t)$$

vendar mi ta ni vrnila zadovoljivih rezultatov. Tudi sicer je profesor navedel 2.10 (kar morda ni najbolj logično, saj bi verjetno v tem primeru napisal 2.1, a ker očitno zna napisati potenco se mi ni zdelo logično, da ne bi znal napisati "*" - kakorkoli že je, funkcija, pomnožena z ustreznim faktorjem se solidno ujema s podatki.

Graf 3 – Md29mn_00001.fio (sinhrotronska meritev)

Pri tej nalogi se GNUplot izkaže v vsej svoji bleščavosti: ne samo, da se da zelo enostavno dodati drugo skalo za program – sproti zna izračunati podatke med stolpci (\ln (prvi stolpec/drugi stolpec)).

Graf izgleda tako:



Graf 3.1 Kljub morebitni nenazornosti sem ohranil "grid" vsaj ene funkcije – v tem primeru rdeče. Zdi se mi, da ne moti preveč – kvečjemu si lahko pomagamo z vzporednicami. Logaritemska skala tu ni tako potrebna, saj bi s tem izgubili nekaj podrobnosti. Zaradi večje nazornosti sem uporabil modro barvo. Verjetno da na abscisni osi niso MeV, vendat naloga ni podala, s kako velikimi energijami imamo opravka.

Opomba: Pred glavo meritev sem na začetek vsake vrstice dodal "#"

Drugo skalo sem naredil tako:

nekje, pred ukazom plot, sem jo "definiral":

```
set y2label "..."  
set y2tics  
set tics out  
set y2range [spodnja_meja:zgornja_meja]
```

s tem sem si pripravil os in vnesel ukaze, ki jih GNUplot sicer vnese avtomatsko, vendar jih moramo tu vnesti sami, da si os pripravimo, kot želimo. Ko želimo to os uporabiti za risanje funkcije vnesemo poleg ukaza plot še ukaz axes x1y2 (uporabimo prvo x os (spodnjo) in drugo y os (desno)):

npr. plot f(x) axes x1y2

Podatke iz šestega, sedmega in osmega stolpca sem prenesel v logaritemsko odvisnost po namigu:

plot ukaz je izgledal nekako tako:

plot "Md2mn_00001.fio" using 1:(log(\$6 / \$7)), "Md29mn_00001.fio" using 1:(log(\$7 / \$8)) axes x1y2
(vmes sem seveda dodal še ukaze za barvo, črte, vendar so tu nepotrebni)

Podobno bi tako lahko spremenil tudi drugo x os (x2).

Navodila sem dobil na internetu na straneh:

<http://www.ibm.com/developerworks/library/l-gnuplot/>

in

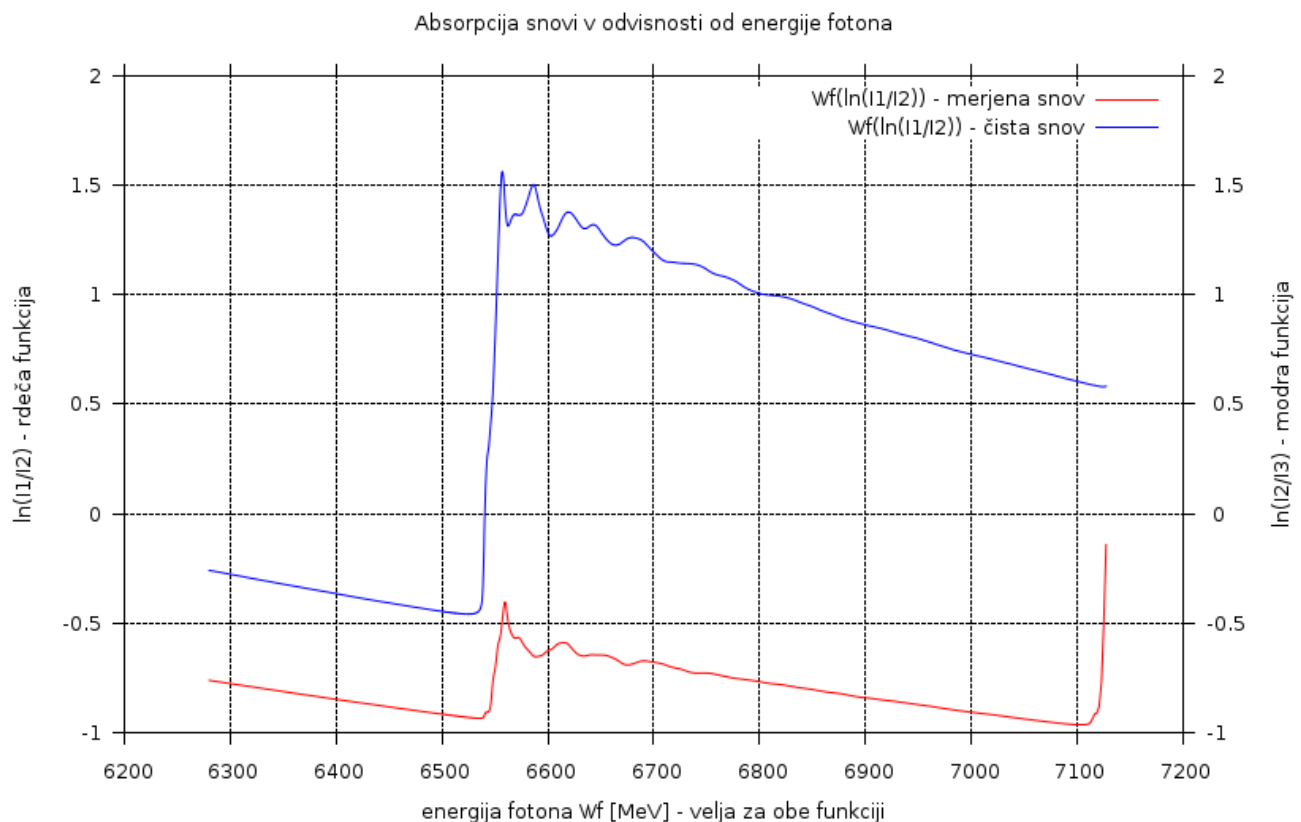
<http://www.duke.edu/~hpgavin/gnuplot.html>

ter še

<http://t16web.lanl.gov/Kawano/gnuplot/index-e.html>

Strani je sicer še več, ampak te dovolj jedrnato in preprostostjo predstavijo rešitev.

Graf bi brez uporabe odatne desne skale bil videti tako:

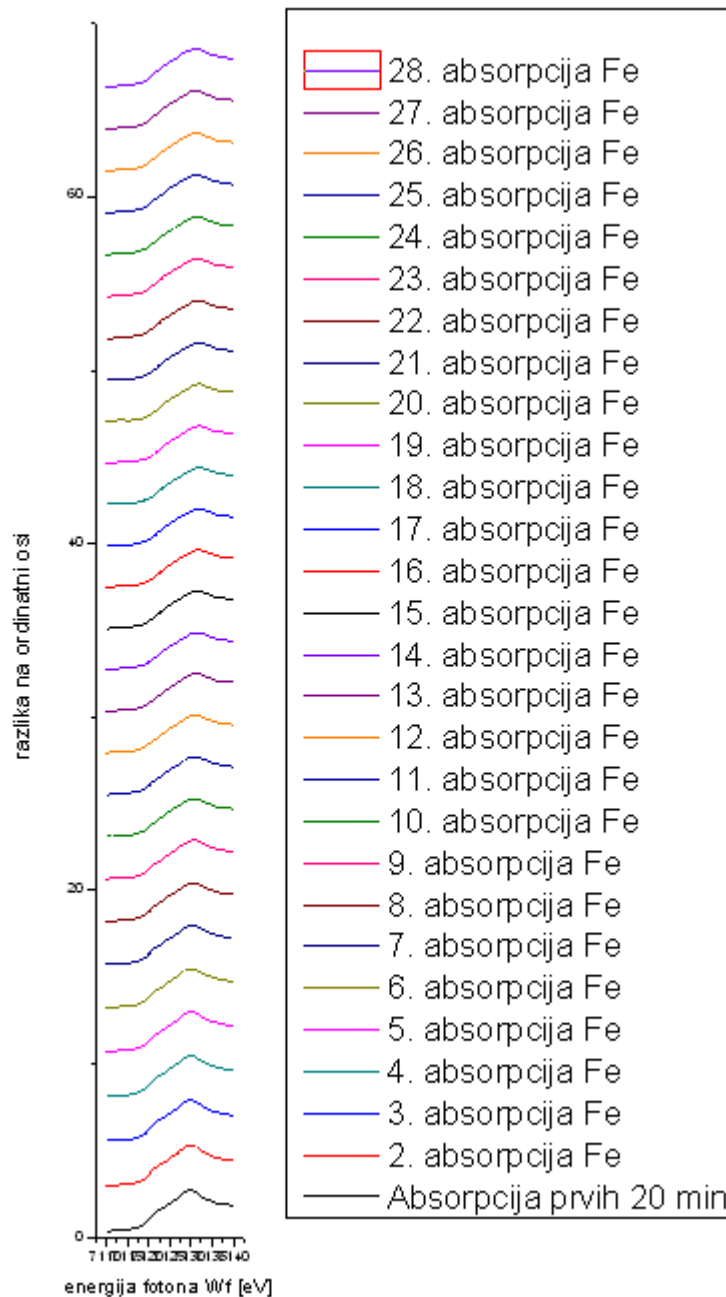


Graf 3.2 Skala je na obeh straneh enaka. Graf je tu zgolj za primerjavo in utemeljitev upravičenosti uporabe grafa z

dvema skalama na vrhu: na prvem grafu (Graf 3.1) je rdeča funkcija veliko bolj jasna in bolj nazorna! Se pa z njo spremeni nekoliko sama predstava o velikosti posameznih funkcij (ne moremo ju obe grafično primerjati, lahko pa obe jasno in različno vidimo). Tudi tu velja, da bi morala biti enota v abscisno os verjetno le eV in ne MeV.

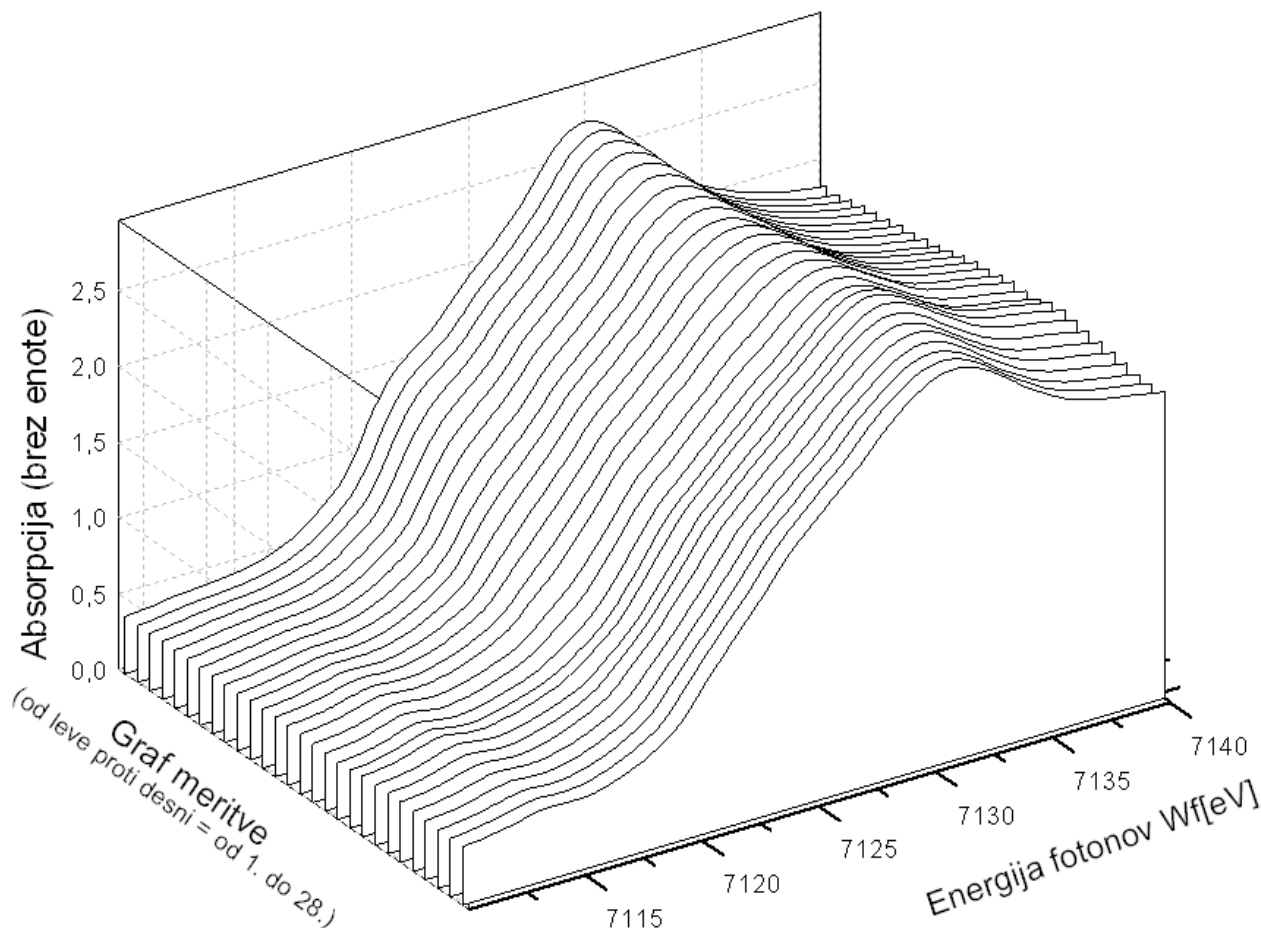
Graf 4 – Fe_rob_0_27.xmu (rentgenski spektri absorpcijskega robu)

Zaradi preprostosti sem ta graf narisal v Origin, ki lahko enostavno z enim ukazom nariše več funkcij, vendar pa zaradi tega ne dopušča toliko svobode.



Graf 4.1 Meritve razmakne po y osi, razliko zapiše v y os (razliko, za koliko je graf dvignjen – kakor da ne gre več od ničle ampak kar naprej). Z risanjem v lomljenih grafih, kjer je grafov več kot 20 so v originu veliki problemi, zato sem uporabil tega. Vendar je morda še najboljša grafična rešitev prikazana spodaj

Graf morda ne prikaže dobre časovne odvisnosti, zato sem dodatl 3D graf, pri čemer je spodnja leva os časovna os, grafi si sledijo v korakih po 20 min.



Graf 4.2 Tako dobimo še 3. dimenzijo – čas, ki je na osi z (graf meritve), kjer je najprej 0.0, nato si grafi sledijo po časovnih intervalih. Iz Takega grafa razberemo, da se krivulja nekoliko umirja. Od časovnje odvisnosti, bi sklepal, da se prične praznjenje pri 20 grafu (tam se prenega polnjenje) – to sklepam po tem, ker ima tisti graf glede na druge maksimum (iz gostote sklepam črt na vrhu sklepam).