

Razvoj jednostavnog algoritma za simulaciju linearnog akceleratora

Barač, Rocco

Undergraduate thesis / Završni rad

2022

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Split, University of Split, Faculty of science / Sveučilište u Splitu, Prirodoslovno-matematički fakultet**

Permanent link / Trajna poveznica: <https://urn.nsk.hr/urn:nbn:hr:166:723189>

Rights / Prava: [Attribution 4.0 International](#)/[Imenovanje 4.0 međunarodna](#)

Download date / Datum preuzimanja: **2024-07-10**

Repository / Repozitorij:

[Repository of Faculty of Science](#)



Sveučilište u Splitu
Prirodoslovno – matematički fakultet

**Razvoj jednostavnog algoritma za simulaciju
linearnog akceleratora**

Završni rad

Rocco Barač

Split, rujan 2022.

Temeljna dokumentacijska kartica

Sveučilište u Splitu
Prirodoslovno – matematički fakultet
Odjel za fiziku
Ruđera Boškovića 33, 21000 Split, Hrvatska

Završni rad

Razvoj jednostavnog algoritma za simulaciju linearnog akceleratora

Rocco Barač

Sveučilišni preddiplomski studij Fizika

Sažetak:

U ovom radu se opisuje i testira jednostavan algoritam za simulaciju rada linearnog akceleratora Alvarezove strukture. Algoritam, uz nekoliko aproksimacija, uspješno simulira rad akceleratora te daje rezultate vrlo slične stvarnima. Simulacije su bitne u fizici jer omogućavaju rješavanje problema za koje ne postoje analitička rješenja, već samo numerička. Ovaj algoritam ne koristi unaprijed definiran akcelerator, već sam odredi kakav akcelerator bi bio optimalan za ubrzavanje zadanih čestica sa zadanim početnim uvjetima. Na kraju rada je uspješno sastavljen DTL dio CERN-ovog Linac4 akceleratora, čije dimenzije su vrlo slične stvarnima te se time pokazuje moguća direktna primjena ovakvog algoritma. Uz daljnji razvoj algoritma i korištenje realnijih elektromagnetskih polja, algoritam bi se mogao koristiti za konstrukciju pravih linearnih akceleratora Alvarezove strukture te za provedbu vrlo preciznih eksperimenata bez potrebe za korištenjem skupe opreme.

Ključne riječi: akcelerator, simulacija, linac, DTL, Alvarez, čestice

Rad sadrži: 19 stranica, 13 slika, 0 tablica, 9 literaturnih navoda. Izvornik je na hrvatskom jeziku.

Mentor: doc. dr. sc. Toni Šćulac

Ocjenjivači: doc. dr. sc. Toni Šćulac,
doc. dr. sc. Petar Stipanović,
dr. sc. Krešimir Dželalija

Rad prihvaćen: 27. 9. 2022.

Rad je pohranjen u Knjižnici Prirodoslovno – matematičkog fakulteta, Sveučilišta u Splitu.

Basic documentation card

University of Split
Faculty of Science
Department of Physics
Ruđera Boškovića 33, 21000 Split, Croatia

Bachelor thesis

Development of a simple algorithm for linear accelerator simulation

Rocco Barač

University undergraduate study programme Physics

Abstract:

The subject of this thesis is the description and subsequent testing of a simple algorithm for simulating an Alvarez type linear accelerator. The algorithm, with a few approximations, successfully simulates the working of accelerators and gives results which are very similar to those of real accelerators. Simulations are important in physics because they enable us to solve problems which don't have analytic solutions, but only numerical ones. This algorithm doesn't use a predefined accelerator, but it instead determines what kind of accelerator would be optimal for accelerating given particles with the given starting conditions. In the end of the thesis the DTL part of CERN's Linac4 accelerator is successfully constructed with dimensions which are very close to the actual ones which demonstrates the possibility of a direct application of this kind of algorithm. With further development of the algorithm, and the use of more realistic electromagnetic fields, the algorithm could be used for the construction of actual linear accelerators of Alvarez type and for carrying out very precise experiments without the need of expensive equipment.

Keywords: accelerator, simulation, linac, DTL, alvarez, particles

Thesis consists of: 19 pages, 13 figures, 0 tables, 9 references. Original language: Croatian.

Supervisor: Assist. Prof. Dr. Toni Šćulac

Reviewers: Assist. Prof. Dr. Toni Šćulac,
Assist. Prof. Dr. Petar Stipanović,
Dr. Krešimir Dželalija

Thesis accepted: Sep 27, 2022

Thesis is deposited in the library of the Faculty of Science, University of Split.

Sadržaj

1	Uvod	1
2	Teorijska pozadina	1
3	Linearni akceleratori	2
3.1	Widerøe linac	2
3.2	Alvarezov linac	3
4	Program	4
4.1	Izbor programskog jezika	4
4.2	Akcelerator i čestice	5
4.3	Algoritam	6
4.3.1	Stvaranje čestica	6
4.3.2	Sastavljanje proizvoljnog akceleratora	6
4.3.3	Električno polje Alvarezovog linaca	7
4.3.4	Gibanje čestica	8
4.3.5	Traženje optimalnog akceleratora	9
5	Simulacije i rezultati	11
5.1	Simulacija sa protonima	11
5.2	Simulacija sa $^{208}\text{Pb}^{53+}$ ionima	13
5.3	Simulacija DTL dijela CERN-ovog Linac4 akceleratora s H^- ionima	15
6	Zaključak	18

1 Uvod

Ideja o nedjeljivim česticama od kojih je izgrađena sva materija potječe još iz davnina. Prvu subatomska česticu, elektron, otkrio je J.J. Thomson krajem 19. stoljeća [1], dok je većina ostalih elementarnih čestica otkrivena u 20. stoljeću. Prirodni izvor elementarnih čestica, a ujedno i najvažniji u prvoj polovici 20. stoljeća, su kozmičke zrake. Visokoenergetske čestice iz svemira prolaskom kroz Zemljinu atmosferu stvaraju takozvane elektromagnetske i hadronske pljuskove, a novonastale čestice mogu se promatrati na površini Zemlje.

Kozmičke zrake nisu idealne za otkrivanje i analiziranje elementarnih čestica jer eksperimenti ovise o prirodnoj pojavi koju je nemoguće kontrolirati. U novije vrijeme elementarne čestice otkrivaju se u akceleratorima čestica, gdje se i ispituju njihova svojstva. Akceleratori čestica ubrzavaju stabilne nabijene čestice poput elektrona, protona, iona te jezgara atoma koristeći elektromagnetska polja. Sudaranjem snopova visokoenergetskih čestica nastaju nove čestice kojima se mogu odrediti svojstva korištenjem raznih detektora, a početni uvjeti su poznati jer se čestice kontrolirano ubrzavaju do poznatih iznosa energije.

Prvi ubrzivači čestica bili su jednostavni uređaji koji bi s jakim elektrostatskim poljem ubrzavali nabijene čestice. Ovakvi uređaji su vrlo ograničeni zbog fenomena električnog proboja (eng. electrical breakdown) tj. prirodnog ograničenja na iznos električnog polja nakon čega neki od izolatora u uređaju počnu provoditi struju. Napredniji tip akceleratora koji koristi alternirajući izvor napona predložio je Gustav Ising 1924. godine [2], a takav uređaj je 1926. godine izgradio Rolf Widerøe [3]. Glavna ideja je da čestice prolaze kroz područja u kojima ih električno polje ubrzava, te kada bi električno polje promijenilo smjer i počelo ih usporavati treba se stvoriti područje koje je izolirano od električnog polja kako se čestica ne bi usporila. Idući veći napredak došao je izumom jačih radio-frekvencijskih izvora napona nakon Drugog svjetskog rata [4] te korištenjem drugačijeg tipa akceleratora kojeg je prvi napravio Alvarez 1946. godine [5].

Ideja ovog rada je opisati razvoj algoritma koji određuje kako složiti optimalan akcelerator za ubrzavanje čestica sa zadanim česticama, početnim uvjetima te maksimalnim naponom unutar područja sa električnim poljem. Algoritam iz ovih varijabli pronalazi odgovarajuće duljine segmenata unutar akceleratora te razmake među njima, nakon čega je moguće koristiti izgrađeni akcelerator za testiranje na većem broju čestica i analiziranje rezultata.

2 Teorijska pozadina

U ovom radu se promatraju čestice visokih energija, stoga je potrebno koristiti relativističke jednadžbe gibanja. Cijelo gibanje se promatra iz laboratorijskog sustava, a pošto se radi o nabijenim česticama u elektromagnetskom polju, promjenu u količini gibanja čestice uzrokuje

Lorentzova sila:

$$\vec{F} = q\vec{E} + q\vec{v} \times \vec{B} \quad (2.1)$$

gdje je \vec{F} 3-sila iz klasične mehanike, q naboj čestice, \vec{v} brzina čestice te su \vec{E} i \vec{B} električno i magnetsko polje u točki prostora u kojoj se čestica nalazi.

Sve perturbacije na sustav se zanemaruju pa je Lorentzova sila jedina sila koju se uzima u obzir. U specijalnoj teoriji relativnosti ne vrijedi vrlo jednostavan oblik 2. Newtonovog zakona gdje je $\vec{F} = m\vec{a}$ nego se za dobivanje akceleracije iz sile koristi jednačba [6]:

$$\vec{a} = \frac{1}{m\gamma} \left(\vec{F} - \frac{(\vec{F} \cdot \vec{v})\vec{v}}{c^2} \right) \quad (2.2)$$

gdje je m masa čestice, $c = 299\,792\,458 \frac{\text{m}}{\text{s}}$ brzina svjetlosti, a $\gamma = \frac{1}{\sqrt{1-\frac{v^2}{c^2}}}$ Lorentzov faktor.

U fizici visokih energija često se koristi kinetička energija čestice umjesto brzine, a u relativističkoj mehanici kinetička energija čestice T dana je izrazom [7]:

$$T = \gamma mc^2 \quad (2.3)$$

3 Linearni akceleratori

3.1 Widerøe linac

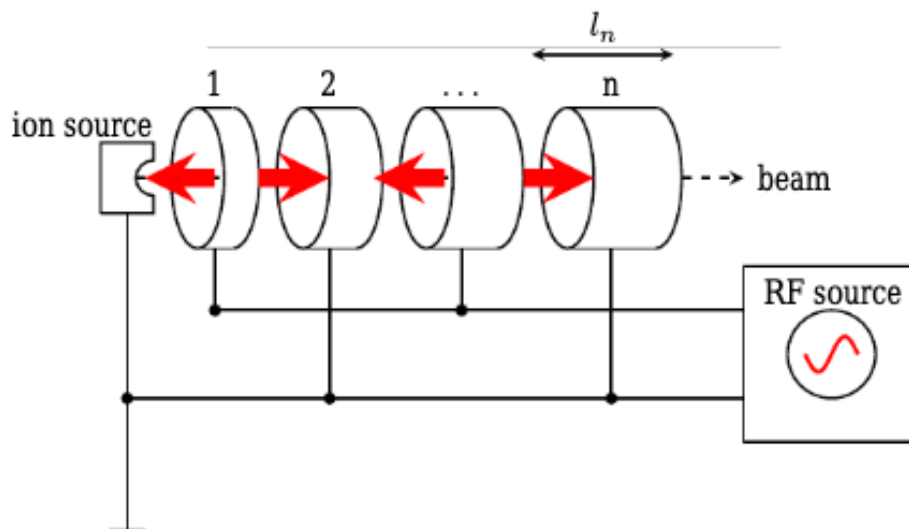
Prvi linearni akcelerator (linac) koji koristi nekoliko područja ubrzavanja i izvor alternirajućeg napona izgradio je Rolf Widerøe 1928. godine [3]. Za stvaranje alternirajućeg napona koriste se radio-frekvencijske (RF) šupljine kojima se mogu postići frekvencije od ~ 10 kHz do 300 GHz. [8] Općenito se ovakav tip linearnog akceleratora gdje čestice prolaze kroz cijevi koje ih štite od vanjskog električnog polja nazivaju Drift Tube Linac (DTL) [4].

Princip rada Widerøe linaca je da se šuplje elektrode kroz koje prolaze čestice postave jedna za drugom tako da su susjedne elektrode spojene na krajeve RF izvora suprotnog polariteta [5]. Na taj način se može postići ubrzavanje čestice u prostoru između susjednih elektroda, dok je polje u šupljini elektroda približno 0 jer su elektrode velike ekvipotencijalne plohe. Iz ovakvog modela akceleratora s RF izvorom napona konstantne frekvencije odmah se vidi da udaljenosti između susjednih procijepa među elektrodama u kojima se ubrzavaju čestice postaju sve veće jer čestica svaki put mora doći u iduće električno polje nakon točno pola perioda RF izvora kako bi se opet ubrzavala, a njena brzina postaje sve veća te se za navedene udaljenosti dobiva

relacija:

$$L_n = \frac{\bar{v}}{2f} \quad (3.1)$$

gdje je \bar{v} prosječna brzina čestice u tom dijelu akceleratora, a f frekvencija RF izvora.



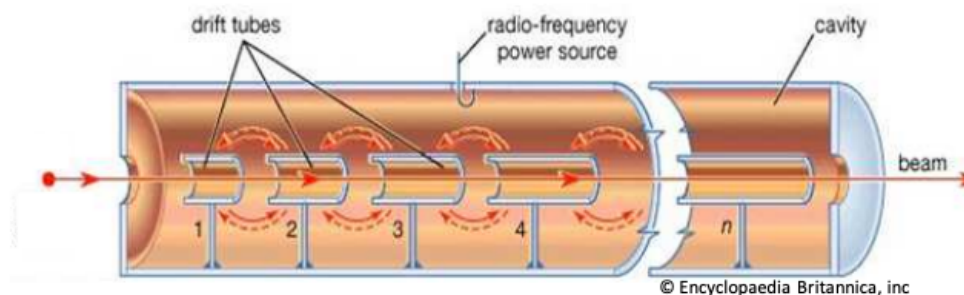
Slika 1: Princip rada Widerøe linaca. Smjerovi RF električnih polja između elektroda označeni su s crvenim strelicama. (slika preuzeta s https://www.researchgate.net/publication/51957233_Cavity_types [5])

Iako je Widerøe linac bio vrlo napredan za svoje vrijeme, ima nekoliko ograničenja. Za veće brzine trebao bi se ili koristiti velik broj jako dugih elektroda, što je vrlo nepraktično, ili povećati frekvenciju RF izvora. Međutim na visokim frekvencijama (>10 MHz) elektrode se počnu ponašati kao antene te sve veći dio energije zrače u obliku radio valova, umjesto da koriste tu energiju za ubrzanje zbog čega bi akcelerator postao vrlo neefikasan. [5]

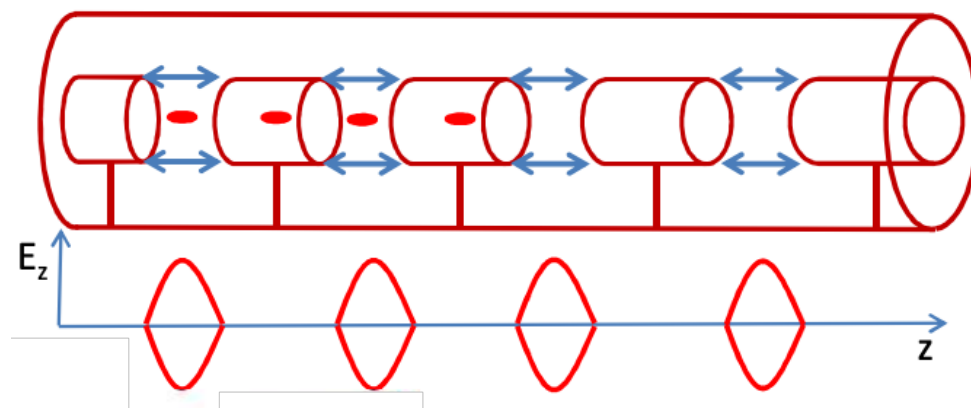
3.2 Alvarezov linac

Problem Widerøe akceleratora pri visokim frekvencijama riješio je Alvarez 1946. godine tako što je stavio Widerøe linac unutar vodljivog cilindra te je cijelu strukturu spojio na RF izvor [5] kao što je prikazano na slici 2. Tip akceleratora kojeg je Alvarez izumio još uvijek se koristi. U ovakvim akceleratorima koriste se RF šupljine za stvaranje stojnih valova elektromagnetskog polja koji ubrzavaju čestice [4]. U navedenoj konfiguraciji električna polja u susjednim procijepima su u fazi, kao što je prikazano na slici 3, što je bitna razlika u odnosu na Widerøe linac gdje su u protufazi.

Kod Alvarezovog tipa linaca čestice se pola perioda izvora ubrzavaju kroz procijep između cijevi, a kad je električno polje svugdje u suprotnom smjeru, koje bi ih usporilo čestice, prolaze kroz cijev koja ih štiti od polja te u tom dijelu provedu drugu polovicu perioda. Iz navedenoga



Slika 2: Shematski prikaz Alvarezovog linaca (slika preuzeta s https://indico.cern.ch/event/532397/contributions/2170633/attachments/1343755/2025070/Alesini_LINEAR_ACCELERATOR_handouts.pdf)



Slika 3: Električno polje Alvarezovog linaca (slika preuzeta s https://indico.cern.ch/event/532397/contributions/2170633/attachments/1343755/2025070/Alesini_LINEAR_ACCELERATOR_handouts.pdf)

se lako vidi da je duljina n -te cijevi l_n dana izrazom:

$$l_n = \frac{v}{2f} \quad (3.2)$$

gdje je v brzina čestice u cijevi koja je u ovom slučaju konstantna, a f frekvencija oscilacije električnog polja.

Određivanje udaljenosti između susjednih cijevi je teže pronaći i u tu svrhu se koristi algoritam koji je opisan u sljedećem poglavlju.

4 Program

4.1 Izbor programskog jezika

Algoritam koji se koristi u ovom radu napisan je u Pythonu. Python omogućava brzo pisanje koda, sadrži brojne biblioteke korisne fizičarima te sadrži klase što omogućuje objektno-orijentirano programiranje. Isti kod može se koristiti za razne akceleratora i čestice

uz samo nekoliko promjena u kodu. Puni kod dostupan je na mom Github profilu¹.

4.2 Akcelerator i čestice

Akcelerator, dijelovi akceleratora (segmenti) i čestice su klase unutar programa. Čestice i segmenti se nakon stvaranja kao objekti prosljeđuju akceleratoru koji ih sprema u liste odakle su lako dostupni osobi koja koristi kod. Neke od čestica definirane su kao potklase, ali lako se može stvoriti bilo kakva čestica. Primjer dijelova koda gdje se uvode klase i potklase prikazani su ispod:

```
39 class particle:      #particle class
40     def __init__(self, q, m, v0, E=[0.,0.,0.], B=[0.,0.,0.], dt=1e-7, dt0=0):
41         self.m = m    #mass
42         self.q = q    #charge
43         self.E = np.array(E)    #electric field
44         self.B = np.array(B)    #magnetic field
45         self.v = np.array([v0])    #velocity
46         self.r = np.array([[0,0,0]])    #position
47         self.t = [0]    #time

214 class proton(particle):      #proton subclass
215     def __init__(self, v0, E=[0.,0.,0.], B=[0.,0.,0.], dt=1e-7):
216         self.m = 1.67262192e-27    #proton mass
217         self.q = 1.60217663e-19    #proton charge
218         super(proton, self).__init__(self.q, self.m, v0, E, B, dt)
219
220 class ion(particle):      #ion subclass
221     def __init__(self, v0, C, A, E=[0.,0.,0.], B=[0.,0.,0.], dt=1e-7):
222         self.m = A*1.66e-27    #ion mass
223         self.q = C*1.60217663e-19    #ion charge
224         super(ion, self).__init__(self.q, self.m, v0, E, B, dt)

229 class segment:      #segment class
230     def __init__(self, l, R, name):
231         self.l = l    #segment length
232         self.R = R    #segment radius (cylindrical segments)

256 class accelerator:      #accelerator class
257     def __init__(self):
258         self.particles = []    #all created particles
259         self.beam = []    #particles which still remain in the beam
260         self.segments = []    #list of segments
```

¹<https://github.com/rbaracl/Bachelor-s-Thesis>

4.3 Algoritam

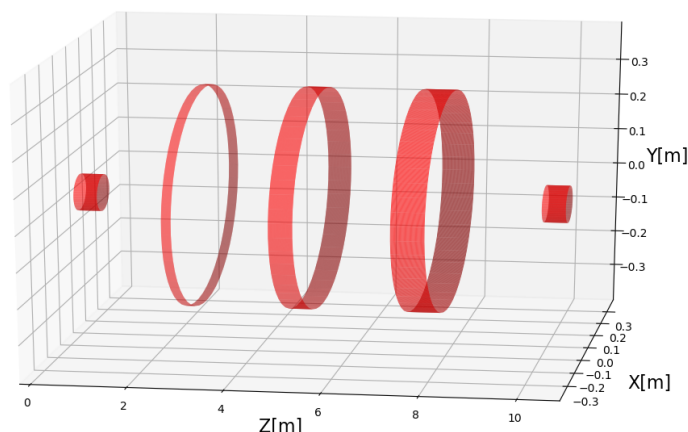
4.3.1 Stvaranje čestica

Čestice se stvore u točki (0,0,0) u trenutku $t=0$. Sustav je postavljen tako da se čestice usmjeravaju u pozitivnom smjeru z-osi koja je ujedno i os simetrije akceleratora. Stvorene čestice ne interagiraju međusobno te postoje devijacije u iznosu i smjeru brzina čestica. Iznos brzine svake čestice dobiva se generiranjem nasumične varijable iz normalne raspodjele sa zadanom prosječnom brzinom i standardnom devijacijom, a smjer čestice se određuje generiranjem dvije nasumične varijable iz konstantne raspodjele vjerojatnosti u zadanim intervalima gdje prva varijabla predstavlja polarni kut, a druga varijabla azimutalni. Stvorene čestice se potom dodaju u akcelerator. Funkcija koja stvara čestice i dodaje ih u akcelerator nalazi se u kodu ispod:

```
21 #create N particles with random directions
22 def create_particles(N, v_ave, v_dev, part="proton", acc=linac, C=0, A=0):
23     for i in range(int(N)):
24         v = np.random.normal(v_ave, v_dev)          #speed
25         theta = np.random.random()*10*np.pi/180    #polar angle
26         fi = np.random.random()*360*np.pi/180     #azimuthal angle
27
28         vx = v*np.sin(theta)*np.cos(fi)
29         vy = v*np.sin(theta)*np.sin(fi)
30         vz = v*np.cos(theta)
31
32         if part=="proton":
33             acc.add_particle(lc.proton([vx, vy, vz]))
34         elif part=="ion":
35             acc.add_particle(lc.ion([vx, vy, vz], C, A))
36         else:
37             acc.add_particle(lc.electron([vx, vy, vz]))
```

4.3.2 Sastavljanje proizvoljnog akceleratora

Prije početka gibanja čestica potrebno je sastaviti sami akcelerator. U kodu je moguće akceleratoru dodavati šuplje cilindre zadanog radijusa i duljine te razmake među njima. Nakon dodavanja svakog novog segmenta i prethodnog definiranja razmaka među njima program spremi položaje z-koordinata početka i kraja dodanog segmenta u liste koje se kasnije koriste za određivanje položaja čestica kako bi se moglo odrediti električno polje koje na njih djeluje te za lakše plotanje segmenata. Proizvoljno generirani akcelerator s 2 uža segmenta na početku i na kraju koji služe kao kolimatori prikazan je na slici 4.



Slika 4: Primjer proizvoljno sastavljenog akceleratora s 2 kolimtora na krajevima i 3 šire cijevi u sredini.

4.3.3 Električno polje Alvarezovog linaca

Za rješavanje jednadžbi gibanja čestica u akceleratoru najprije je potrebno odrediti iznos električnog polja u svim točkama prostora. U unutrašnjosti cijevi čestice su zaštićene od utjecaja bilo kakvog električnog polja. Za područja između susjednih cijevi u ovom algoritmu koristi se linearna aproksimacija tj. aproksimacija da potencijal raste ili pada linearno od kraja jedne elektrode do početka iduće te da je smjer električnog polja paralelan sa z-osi. Ako se odabere da je potencijal 0 na polovici procijepa između 2 cijevi onda je potencijal V oblika:

$$V(z, t) = \text{sgn}(q) \cdot 2 \frac{z - z_0 - \frac{d}{2}}{d} V_{\max} \cos(\omega(t - t_0)) \quad (4.1)$$

gdje funkcija $\text{sgn}(q)$ vraća predznak naboja čestice, $z - z_0$ je udaljenost od kraja prethodne cijevi, d je udaljenost između dvije cijevi tj. širina procijepa, V_{\max} je maksimalni iznos potencijala u šupljini, a član $\cos(\omega(t - t_0))$ se pojavljuje zbog korištenja RF izvora oscilirajućeg napona kutne frekvencije $\omega = 2\pi f$. Član t predstavlja vrijeme od početka gibanja čestica, a t_0 određuje algoritam s ciljem da što veći broj čestica upadne u prostor između prve dvije cijevi u trenutku $t = t_0$ jer je u tom trenutku vrijednost kosinusa maksimalna što znači da se iza tog trenutka potencijalna energija pretvara u kinetičku što je i cilj.

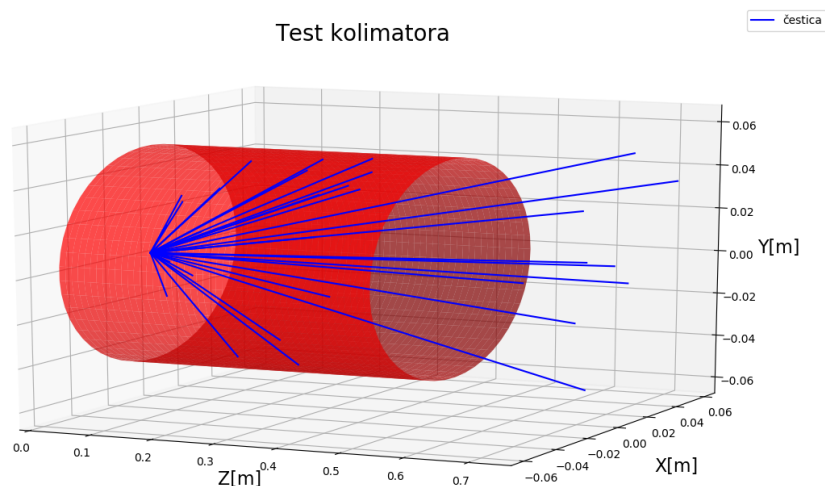
Za električno polje između elektroda uzima se negativni gradijent potencijala te se konačno dobiva:

$$E(t) = \text{sgn}(q) \frac{2}{d} V_{\max} \sin(\omega(t - t_0)) \quad (4.2)$$

4.3.4 Gibanje čestica

Za gibanje čestica koristi se Runge-Kutta 4 (RK4) metoda. Prije pozivanja funkcije koja koristeći RK4 metodu pomakne česticu u jednom vremenskom trenutku, potrebno je provjeriti gdje se čestica nalazi jer o tome ovisi postoji li električno polje i može li se čestica uopće pomaknuti ili je udarila u barijeru u prethodnom koraku. U tu svrhu koriste se dva brojača od kojih jedan broji koliko je puta čestica ušla u novi segment, a drugi koliko je puta izašla. Ako je čestica više puta ušla u neki segment akceleratora nego što je izašla to znači da se čestica nalazi unutar cijevi te da je električno polje u tom području jednako 0. Pošto u ovom algoritmu nisu upaljena nikakva magnetska polja, iz jednadžbi 2.1 i 2.3 se odmah vidi da je akceleracija čestice u toj točki također 0. Ovaj brojač se koristi i za provjeru ulaska čestice u novi segment te za uzimanje radijusa segmenta u kojem se čestica nalazi. Kada je brojač početaka i krajeva segmenata jednak n , što znači da je čestica izvan cijevi i dotad ih je prošla n , onda program u svakom koraku provjerava je li čestica prešla z -koordinatu početka n plus prvog segmenta koristeći listu u kojoj su pohranjene z -koordinate početaka segmenata koristeći brojač početaka kao indeks. U slučaju da je čestica ušla u područje nove cijevi, brojač početaka se povećava za 1, čime taj brojač postaje veći od brojača krajeva te se ispunjava uvjet da je akceleracija u svakom idućem koraku jednaka 0 sve dok se brojač krajeva također ne poveća. Ispunjavanjem navedenog uvjeta za ulazak u cijev program sada u svakom koraku također provjerava je li čestica udarila u rub segmenta u kojem se trenutno nalazi te uklanja česticu iz liste snopa čestica čime njeno gibanje završava u slučaju da je taj uvjet ispunjen. U svakom koraku se također provjerava je li čestica došla do kraja segmenta te ako je, brojač zaustavljanja se povećava za 1 što ispunjava uvjet da električno polje više nije 0 nego je zadano jednadžbom 4.2. Vrijeme, položaj i brzinu čestica potrebno je spremati u liste u svakom koraku gibanja jer se koriste u drugim dijelovima algoritma.

Jedan od problema koji se pojavljuje pri implementaciji algoritma za gibanje čestica je velik raspon redova veličine brzina čestica za vrijeme gibanja. Na kraju gibanja čestice mogu biti ubrzane gotovo do brzine svjetlosti, dok im na početku gibanja brzina može biti manja i za 4-5 redova veličine. Ako se pri numeričkom rješavanju jednadžbi gibanja koristi premali vremenski pomak Δt onda je gibanje precizno za velike brzine, ali se program vrlo sporo pokreće jer je potreban jako velik broj koraka dok se čestica ne ubrza na neke veće brzine što je potpuno nepraktično jer bi se za jedno izvršavanje programa trebalo čekati možda i po nekoliko dana. S druge strane, ako je vremenski korak prevelik program postaje vrlo neprecizan za velike brzine što, osim nepreciznosti, može dovesti i do errora jer čestice mogu prijeći brzinu svjetlosti čime se u Lorentzovom faktoru u jednadžbi 2.3 pojavi negativan broj pod korijenom. Da bi se oba problema istovremeno izbjegla uveden je varijabilni vremenski korak koji se mijenja ovisno o redu veličine brzine čestice. Algoritam pri malim brzinama uzima puno veći Δt nego pri velikim što za posljedicu ima razumno vrijeme pokretanja programa kao i vrlo precizne rezultate. Na slici 5 prikazan je primjer prolaska stvorenih čestica kroz kolimator.



Slika 5: *Primjer prolaska čestica kroz kolimator.*

4.3.5 Traženje optimalnog akceleratora

Prvi korak u traženju optimalnog akceleratora je pitanje koju česticu se treba najviše ubrzati. Ne može se samo uzeti prosječna brzina stvorenih čestica jer čestice imaju komponente brzine i u x i y smjeru što znači da bi prosječna čestica na početku trebala imati manju brzinu u z-smjeru od one koja je zadana za generiranje nasumičnih iznosa brzine što se može vidjeti iz koda priloženog u potpoglavlju 4.3.1. Način na koji se nađe odgovarajuća čestica je da se prvo stvori prvi kolimator akceleratora te se pusti snop čestica prosječne brzine i devijacije koja će se koristiti i u daljnjim simulacijama. Električna polja su u ovoj fazi ugašena te se uzme medijan z komponenti brzina čestica koje su izašle iz kolimatora. Čestica brzine medijana brzina je upravo ona koja će se najbolje ubrzati u ovom algoritmu.

Nakon dobivanja medijana brzina čestica, sve čestice se uklanjaju iz akceleratora te započinje postupak dodavanja cijevi odgovarajućih duljina na odgovarajuće položaje akceleratora. Za početak se stvori nova test čestica čija je z komponenta brzine jednaka medijanu z komponenti brzina iz prethodnog testa, a x i y komponente brzine su 0 kako se test čestica ne bi zabila u rub cijevi ili kolimatora. Čestica se pusti da dođe do kraja kolimatora, nakon čega se doda prva iduća cijev proizvoljne duljine te prvi procijep duljine d koju bi u pola perioda izvora prešla čestica koja se giba nekom konstantnom brzinom iznosa većeg od početne brzine test čestice, a manjeg od brzine svjetlosti jer će iznos prosječne brzine test čestice biti upravo u tom intervalu. Nakon uzimanja nekakve sredine (npr. aritmetičke) između početne brzine test čestice i brzine svjetlosti te računanja prvog pokušaja d , test čestica se pusti kroz procijep te se u svakom koraku gibanja čestice provjerava je li procijep predug, prekratak ili odgovarajuće dužine. Uvjet da je procijep prekratak se ispuni kada čestica prođe cijeli procijep i uđe u cijev prije nego što je električno polje promijenilo predznak te počelo usporavati česticu, a uvjet da je predug se ispuni ako električno polje promijeni predznak prije nego što čestica pređe cijeli

procijep. Ovakvi uvjeti logički imaju smisla, ali dovode do beskonačnih petlji pri pokretanju programa ako se samo unesu u program bez dodatnih modifikacija. Kod numeričkog rješavanja problema gibanje je skup velikog broja pomaka konačne duljine te je skup mogućih duljina koraka konačan jer računalo ne može računati s bilo kojim realnim brojem već ima ograničen broj bitova memorije za svaku varijablu što mu daje konačnu preciznost. Iz ovih činjenica se odmah vidi da je nemoguće naći egzaktnu vrijednost duljine procijepa već moramo definirati što je prihvatljivo blizu. Ako čestica prijeđe cijeli procijep dok je električno polje još uvijek ubrzavalo na kraju procijepa, duljina procijepa se prihvaća ako je polje već jako slabo u odnosu na maksimalan iznos polja. Također, ako polje počne usporavati česticu udaljenost procijepa se prihvaća ako se čestica već nalazi vrlo blizu početka iduće cijevi. U slučaju da je procijep bio predug, duljina procijepa se skрати množenjem nekim brojem koji je veći od 0, a manji od 1 te ako je bio prekratak, duljina mu se množi s brojem većim od 1 s tim da se mora pripaziti da ova 2 broja nisu recipročna kako bi se izbjegla beskonačna petlja. Primjer navedenog procesa unutar petlje u kojoj čestica radi korake prikazan je u pseudokodu priloženom ispod:

```
1 # d - duljina procijepa
2 # Emax - maksimalni iznos elektricnog polja
3 # E - trenutni iznos elektricnog polja
4 # Eprev - iznos elektricnog polja u prethodnom koraku
5 # q - naboj cestice
6 # z - udaljenost cestice od pocetka procijepa
7
8 if (|E/Emax|<0.01 and |z/d|<0.01):
9 {
10     print("Procijep je odgovarajuće duljine!")
11 }
12
13 else if (E*q<0 and Eprev*q>0):
14 {
15     print("Procijep je prekratak!")
16     d *= 1.1
17 }
18
19 else if z>d:
20 {
21     print("Procijep je predug!")
22     d *= 0.9
23 }
```

Ako duljina procijepa nije pogođena iz prve, duljina se modificira te se čestica se vrati nazad na početak procijepa. Ovo se lako napravi ako se uvede brojač koraka koji počinje od 0 na početku svakog procijepa te se ukloni onoliko zadnjih elemenata lista vremena, položaja i brzina čestice koliko je brojač koraka prebrojao, nakon čega se brojač resetira i cijeli proces kreće ispočetka dok se ne pogodi odgovarajuća duljina procijepa. Nakon što čestica prođe kroz

prvi procijep, duljina cijevi se promijeni u odgovarajuću duljinu korištenjem trenutne brzine čestice i uvrštavanjem u jednadžbu 3.2 te puštanjem čestice da prođe kroz cijev i dođe do idućeg procijepa gdje se računanje duljine procijepa opet odvija koristeći gore navedeni algoritam. Cijeli ovaj postupak se prekida kada se uspješno izgradi akcelerator od onoliko cijevi koliko je korisnik unio u funkciju prije pokretanja programa.

Za provjeru postupka pronalaženja duljine procijepa iz jednadžbe 2.3 se može izvesti izraz kojeg mora zadovoljavati brzina čestice na kraju procijepa. U procijepu su i brzina i sila u pozitivnom z smjeru. Izlučivanjem sile iz zagrade izraz koji ostaje u zagradi jednak je γ^{-2} . Ako se akceleracija zapiše kao dv/dt i uvrsti sila koristeći jednadžbe 4.2 i 2.1 dobiva se sljedeća diferencijalna jednadžba:

$$\frac{dv}{dt} = \frac{q}{m} E(t) \gamma(v)^{-3} \quad (4.3)$$

koja se može riješiti separacijom varijabli. Jednadžba se prvo prebaci u oblik:

$$\left(1 - \frac{v^2}{c^2}\right) dv = \frac{2|q|}{md} V_{\max} \sin(\omega(t - t_0)) dt \quad (4.4)$$

gdje se integriranjem obe strane jednadžbe dobije analitičko rješenje. Granice integrala po brzini mogu se označiti sa v_i i v_f , a granice integrala po vremenu sa t_i i t_f . Kosinusi koji se dobiju integracijom po vremenu lako se srede jer čestica dođe na početak prvog procijepa kad je $t = t_0$ te na početak n -tog idućeg procijepa nakon $t = t_0 + n \cdot T$ pa za sve vrijednosti t_i vrijedi $\cos(\omega(t_i - t_0)) = 1$. Na kraj svakog procijepa čestica dođe pola perioda nakon što dođe na početak pa se odmah vidi da vrijedi $\cos(\omega(t_f - t_0)) = -1$. Uvrštavanjem svega ovoga u integral jednadžbe 4.4 dobiva se izraz za konačnu brzinu v_f :

$$v_f \gamma(v_f) = v_i \gamma(v_i) + \frac{4|q|V_{\max}}{\omega md} \quad (4.5)$$

Numerički rezultat se sada može provjeriti koristeći analitički rezultat tako što mora vrijediti:

$$\frac{v_i \gamma(v_i) + \frac{4|q|V_{\max}}{\omega md}}{v_f \gamma(v_f)} \approx 1 \quad (4.6)$$

5 Simulacije i rezultati

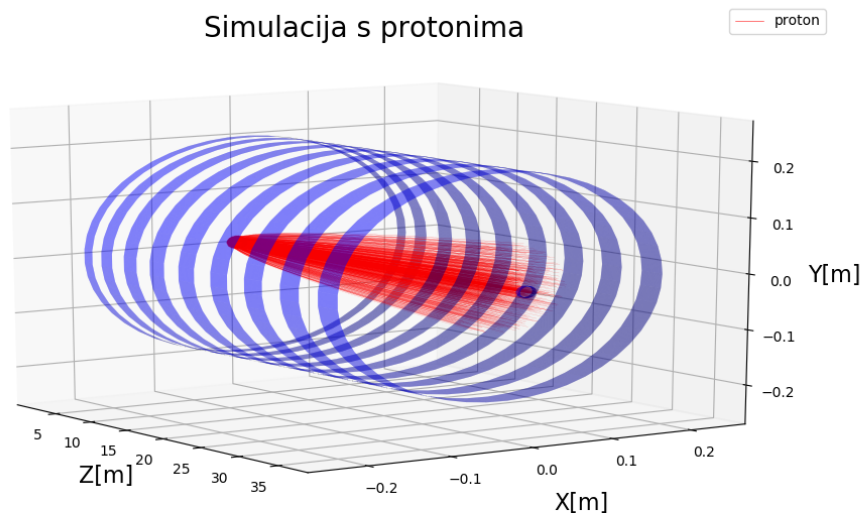
5.1 Simulacija sa protonima

Prva simulacija na kojoj se testira algoritam je test s protonima. Akcelerator koji se treba sastaviti ima 2 identična kolimatora na početku i na kraju duljine 50 cm i radijusa 1 cm te 9 cijevi sredini. Električno polje oscilira frekvencijom 300 MHz, a V_{\max} iznosi 50 MV. Čestice koje se generiraju imaju prosječni iznos brzine 0,08 c sa standardnom devijacijom 100 puta

manjeg iznosa. Maksimalni kut početne brzine čestica u odnosu na z-os (polarni kut) je 10° .

Puštanjem test čestice dobivaju se rezultati koji se slažu s analitičkim rješenjem iz jednadžbe 4.6. Test čestica se ubrzala od 0,0802 c do 0,801 c. Udaljenost između sredina prva 2 procijepa je 206 cm, a između sredina zadnja 2 procijepa je 494 cm što ima smisla s obzirom na to da čestica postaje sve brža kroz simulaciju.

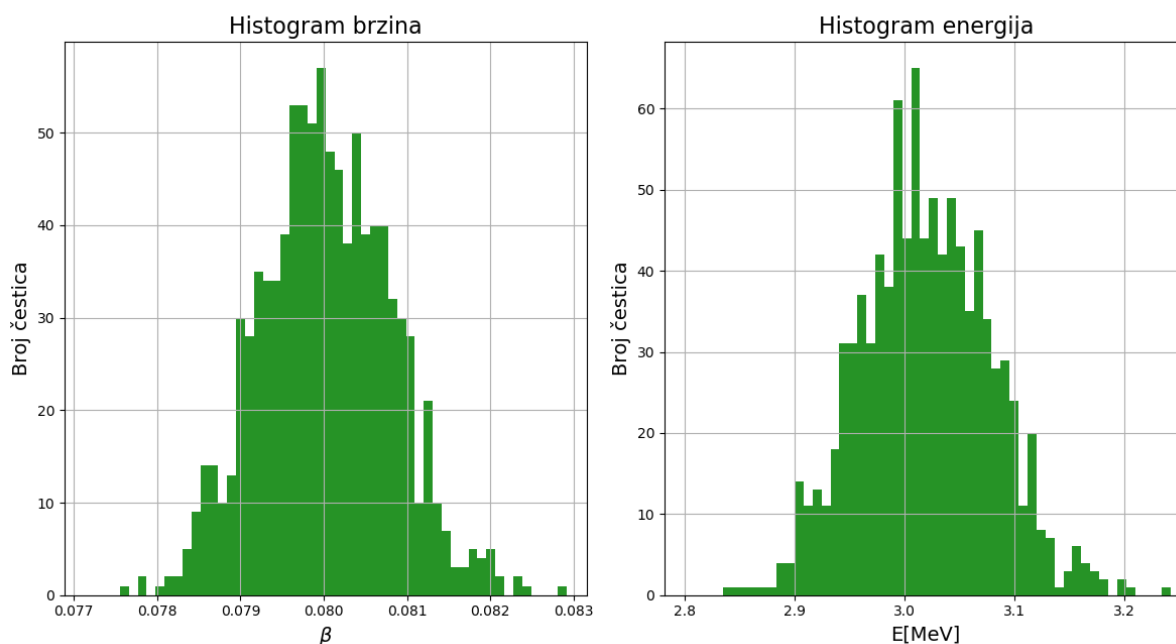
Nakon uspješnog sastavljanja akceleratora kroz akcelerator je pušten snop od 10.000 protona. Kroz prvi kolimator prošle su 952 čestice prosječne brzine 0,0800 c sa standardnom devijacijom iznosa 0,0008 c što odgovara energiji $(3,02 \pm 0,06)$ MeV. Nakon prolaska do kraja akceleratora, iz drugog kolimatora izađe 125 čestica brzine $(0,801 \pm 0,001)$ c što odgovara energiji (629 ± 4) MeV. Navedeni rezultati prikazani su na slikama 7 i 8 gdje je $\beta \equiv \frac{v}{c}$, a slika simulacije gdje protoni još prolaze kroz cijevi prikazana je na slici 6. Raspodjela brzina i energija nakon 1. kolimatora ima oblik gausijana kao što je i očekivano jer se čestice nisu filtrirale ovisno o iznosu početne brzine već samo o smjeru. Asimetrija u raspodjeli brzina i energija nakon 2. kolimatora također ima smisla jer se čestice bliže prosječnoj brzini nakon prvog kolimatora najviše ubrzavaju, a one koje su i brže i sporije se ne ubrzavaju optimalno pa su na kraju čestice koje su bile brže od prosjeka također postale sporije od čestica koje su pri izlasku iz 1. kolimatora bile blizu prosječne brzine.



Slika 6: Prolazak 10.000 protona kroz akcelerator.

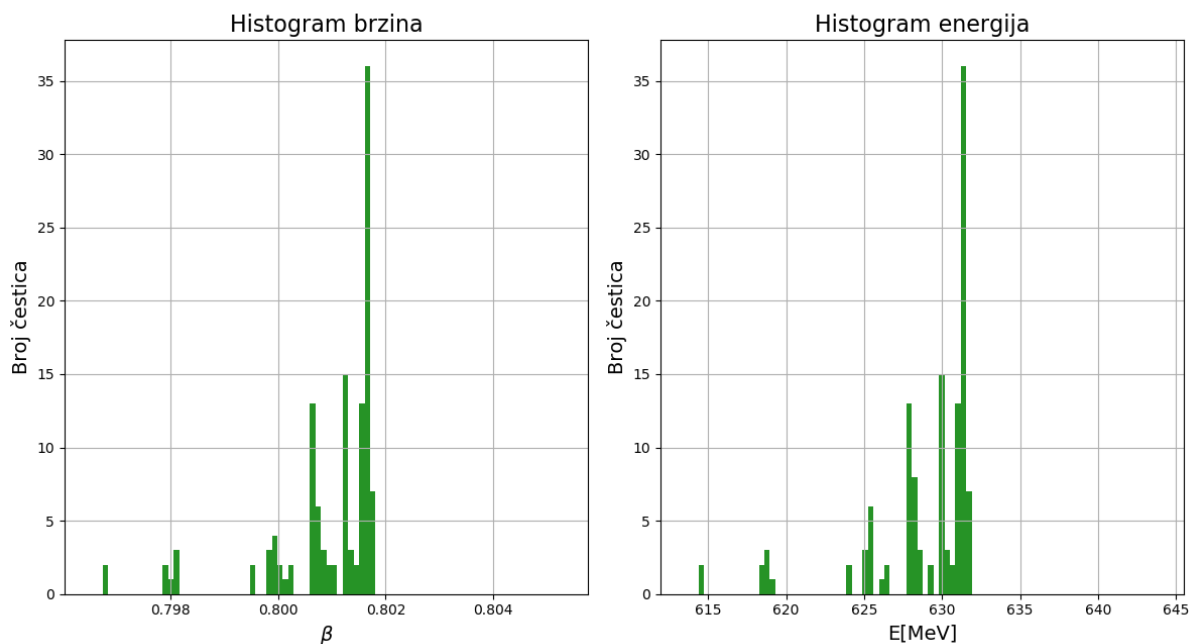
Gubitak ovako velikog broja čestica je očekivan s obzirom na to da u algoritmu nema nikakvog mehanizma za fokusiranje snopa čestica osim kolimatora koji uklone čestice iz simulacije.

1. kolimator



Slika 7: Raspodjela brzina i energija čestica nakon prolaska kroz 1. kolimator u simulaciji s 10.000 protona. Kroz prvi kolimator prošle su 952 čestice.

2. kolimator



Slika 8: Raspodjela brzina i energija čestica nakon prolaska kroz 2. kolimator u simulaciji s 10.000 protona. Kroz prvi kolimator prošlo je 125 čestica.

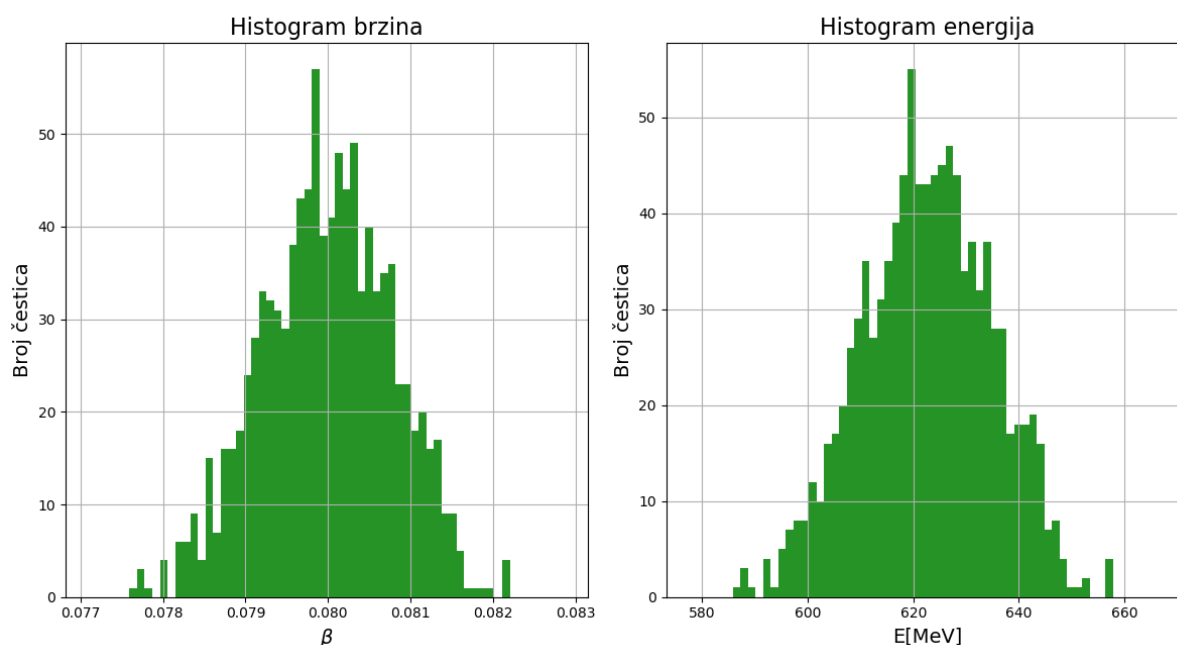
5.2 Simulacija sa $^{208}\text{Pb}^{53+}$ ionima

Iduća simulacija koristi sve iste početne uvjete iz prošle simulacije, ali sada se koriste $^{208}\text{Pb}^{53+}$ ioni umjesto protona.

Puštanjem test čestice dobivaju se rezultati koji se slažu s analitičkim rješenjem iz jednadžbe 4.6. Test čestica se ubrzala od 0,0798 c do 0,523 c. Udaljenost između sredina prva 2 procijepa je 119 cm, a između sredina zadnja 2 procijepa je 317 cm. Analiziranjem jednadžbe 4.5 vidi se da konačna brzina na kraju svakog procijepa ovisi o omjeru $\frac{|q|}{m}$ koji je u ovoj simulaciji oko 4 puta manji nego u prošloj pa je očekivano da se ioni $^{208}\text{Pb}^{53+}$ manje ubrzaju nego protoni što za posljedicu nosi da su razmaci između sredina procijepa veći.

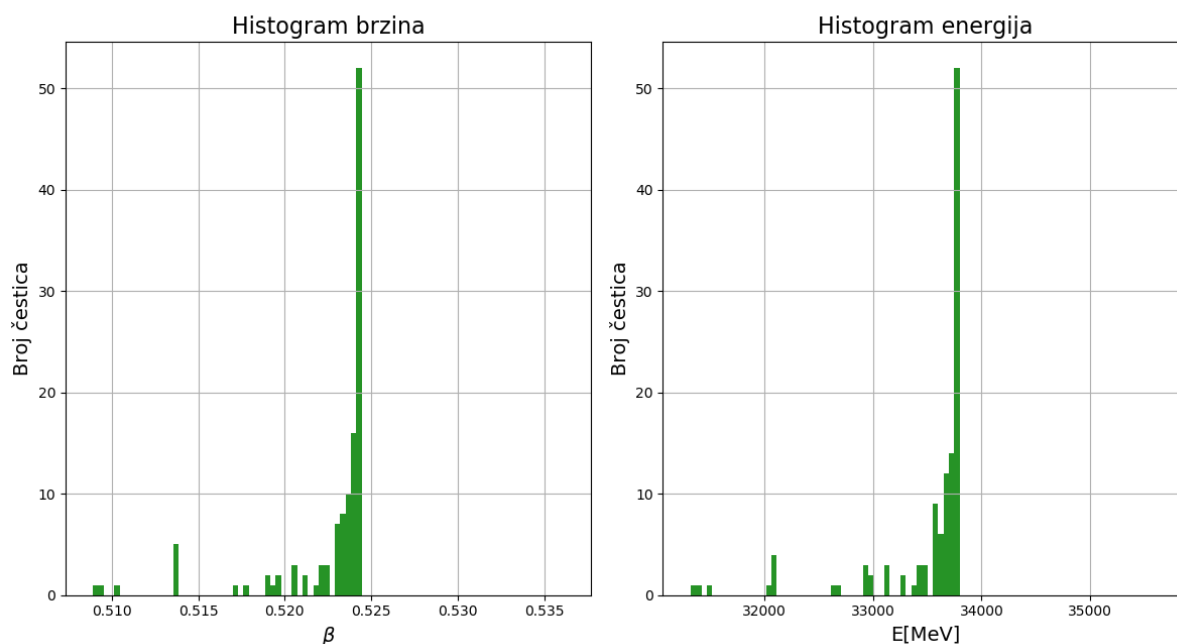
Nakon uspješnog sastavljanja akceleratora kroz akcelerator je pušten snop od 10.000 $^{208}\text{Pb}^{53+}$ iona. Kroz prvi kolimator prošlo je 1011 čestica prosječne brzine 0,0800 c sa standardnom devijacijom iznosa 0,0008 c što je identičan rezultat kao i u prošloj simulaciji, ali sada to, zbog znatno veće mase, odgovara energiji (620 ± 10) MeV. Nakon prolaska do kraja akceleratora, iz drugog kolimatora izađe 121 čestica brzine $(0,523\pm 0,004)$ c što odgovara energiji $(33,5\pm 0,6)$ GeV. Navedeni rezultati prikazani su na slikama 9 i 10. Raspodjela brzina i energija je simetrična nakon 1. kolimatora, a nakon 2. kolimatora je asimetrična s većinom čestica lijevo od maksimuma kao što je i očekivano, a razlog toga je objašnjen u prošlom primjeru.

1. kolimator



Slika 9: Raspodjela brzina i energija čestica nakon prolaska kroz 1. kolimator u simulaciji s 10.000 $^{208}\text{Pb}^{53+}$ iona. Kroz prvi kolimator prošlo je 1011 čestica.

2. kolimator



Slika 10: Raspodjela brzina i energija čestica nakon prolaska kroz 2. kolimator u simulaciji s 10.000 $^{208}\text{Pb}^{53+}$ iona. Kroz prvi kolimator prošla je 121 čestica.

5.3 Simulacija DTL dijela CERN-ovog Linac4 akceleratora s H^- ionima

CERN-ov Linac4 za dio ubrzavanja koristi DTL Alvarezove strukture kojim ubrza negativne ione vodika od brzine 0,08 c do 0,31 c tj. od energije 3 MeV do 50 MeV kroz 120 cijevi koristeći frekvenciju izvora od 352 MHz [9] [4]. Pritom udaljenosti između sredina susjednih procijepa variraju od 68 mm do 264 mm [4].

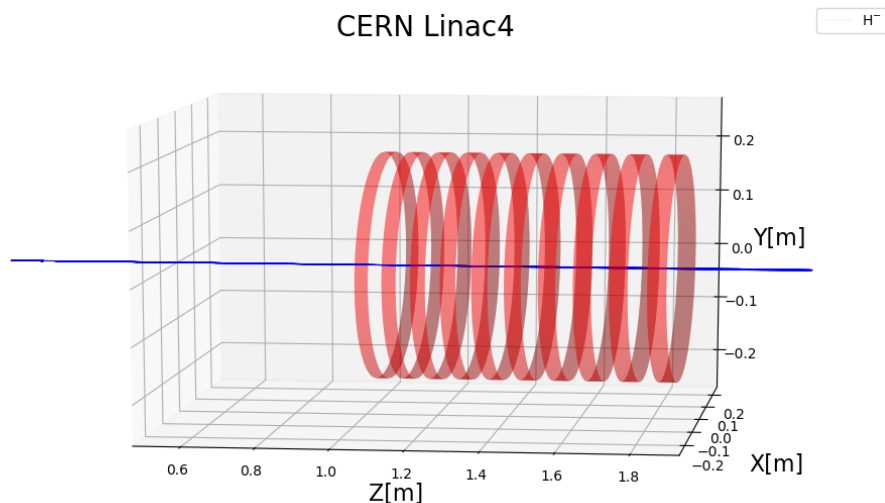
Za ovu simulaciju postavljanje varijable $V_{\max} = 300\text{kV}$ daje rezultate vrlo slične stvarnima tj. čestice brzine $\sim 0,08c$ se ubrzaju do $\sim 0,31c$.

Test čestica i konstrukcija akceleratora za ovu simulaciju uspješno su izvršeni. Rezultati se slažu s analitičkim rješenjem iz jednadžbe 4.6 i test čestica se ubrza od 0,08 c do 0,31 c s razmacima između sredina susjednih procijepa koji variraju od 76 mm do 263 mm što je vrlo slično vrijednostima pravog akceleratora.

Ovu simulaciju je teže izvesti s algoritmom koji se koristi jer je akcelerator puno duži pa se znatno veći broj čestica izgubi te se stoga varijable moraju pažljivo birati kako bi se dobili korisni podaci u razumnom vremenu izvršavanja programa. Prosječna brzina iznosi 0,08 c sa standardnom devijacijom gausijana 1000 puta manjeg iznosa dok je maksimalni polarni kut 1° . Početni i konačni kolimator dugi su 1 m, a radijus im iznosi samo 1 mm.

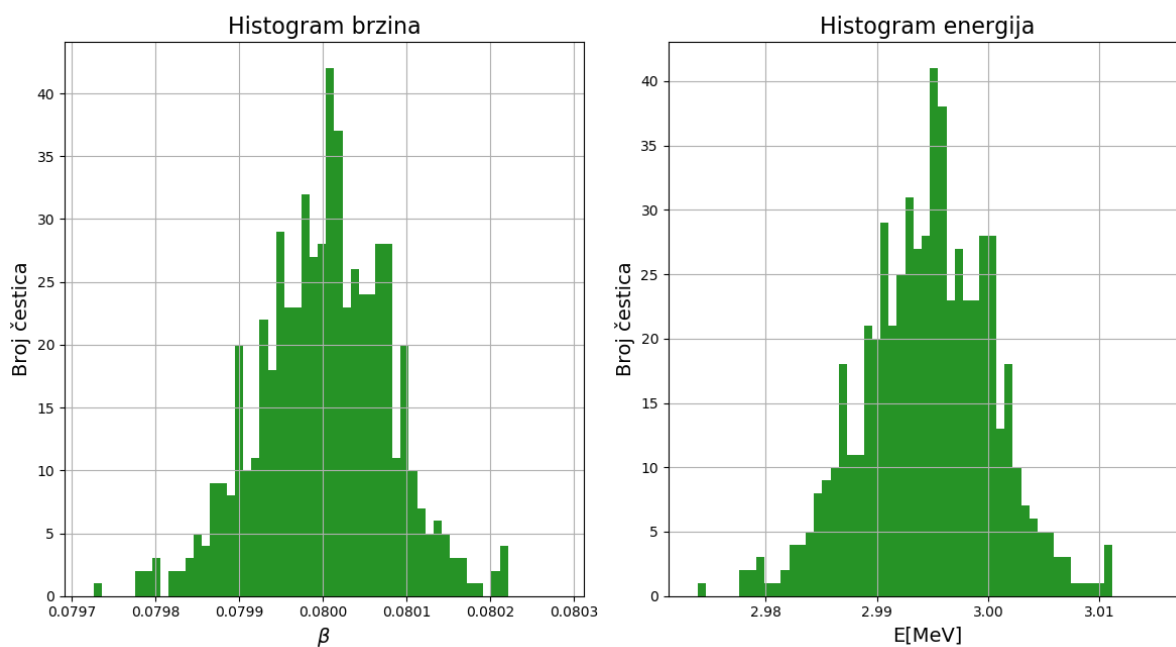
Nakon sastavljanja Linac4 akceleratora simulacija se pokrene sa 10.000 negativnih iona vodika. Kroz prvi kolimator prošlo je 604 čestica prosječne brzine 0,08000 c sa standardnom devijacijom iznosa 0,00008 c što odgovara energiji $(2,995 \pm 0,006)$ MeV. Na kraju simulacije iz

drugog kolimatora izašlo je samo 45 čestica brzina $(0,26 \pm 0,07)$ c i energija (40 ± 20) MeV. Dio simulacije prikazan je na slici 11, a histogrami rezultata prikazani su na slikama 12 i 13.



Slika 11: Dio simulacije DTL dijela CERN-ovog Linac4 akceleratora s 10.000 H⁻ iona.

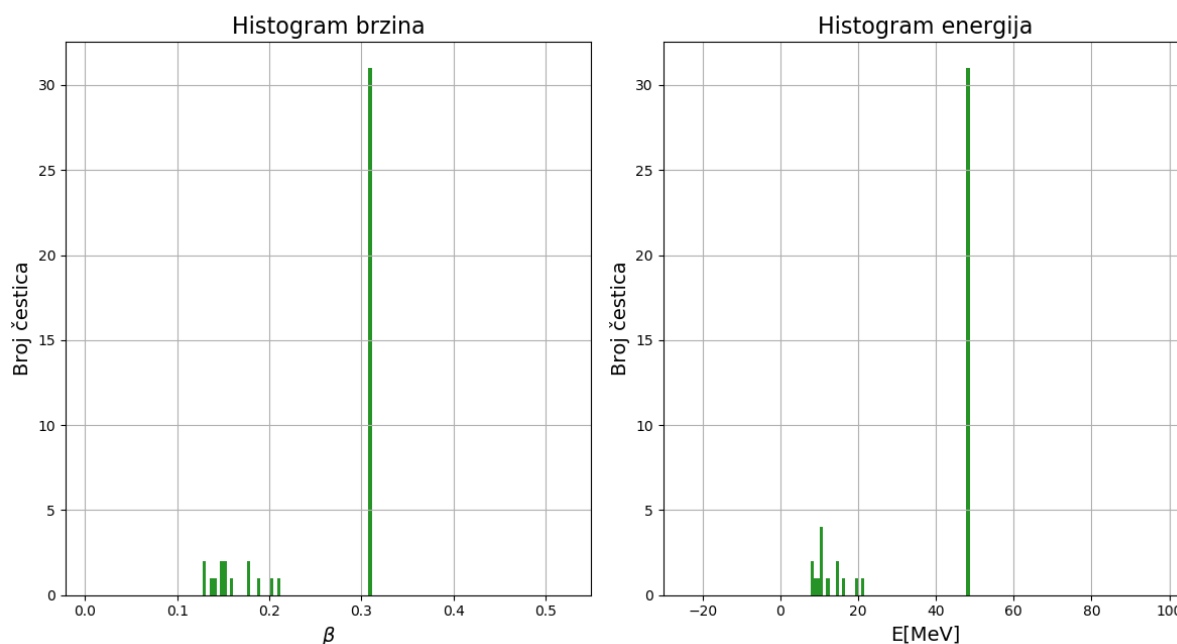
1. kolimator



Slika 12: Raspodjela brzina i energija čestica nakon prolaska kroz 1. kolimator u simulaciji s 10.000 H⁻ iona. Kroz prvi kolimator prošle su 604 čestice.

Analiziranjem histograma na slici 13 vidi se da su rezultati zapravo bolji nego što se čine ako se samo pogleda prosječna vrijednost energije i brzine snopa čestica na kraju simulacije.

2. kolimator



Slika 13: Raspodjela brzina i energija čestica nakon prolaska kroz 2. kolimator u simulaciji s 10.000 H^- iona. Kroz prvi kolimator prošlo je 45 čestica.

Velika većina čestica od onih koje su prošle kroz drugi kolimator ubrzala se do $\sim 0,31 c$, kao što su i trebale, a ostale čestice su zapele na puno nižim vrijednostima brzine pa spuštaju prosjek. Ovaj akcelerator je bio puno duži od onih iz prethodnih primjera pa se ovakav rezultat mogao i očekivati.

Ovim primjerom je pokazano da se s relativno jednostavnim algoritmom mogu dobiti smisleni rezultati čak i za simulacije stvarnih akceleratora. Najveći problem s ovako velikim akceleratorima je da snop mora biti fokusiran jer inače dolazi do gubitka velike većine čestice. U stvarnosti se u DTL dio CERN-ovog Linac4 akceleratora injektira već fokusirani snop koji se ubraza u prethodnom dijelu akceleratora i snop se drži fokusiranim koristeći kvadropolne magnetne [9].

6 Zaključak

U ovom radu je pokazano da se s relativno jednostavnim algoritmom mogu simulirati linearni akceleratori čestica. Rezultati iz prethodnog poglavlja slažu se s analitičkim predviđanjima iz jednadžbe 4.6, a sam algoritam je vrlo fleksibilan jer se lako može primijeniti na bilo koju nabijenu česticu. Sve simulacije koje su analizirane u ovom radu daju očekivane oblike distribucija brzina i energija te iznose brzina na početku i na kraju simulacije koje se slažu s analitičkim predviđanjima. Prije ulaska u električno polje energije čestica zadrže normalnu distribuciju, a na kraju se pojavljuje asimetrična distribucija zbog optimalnog ubrzavanja čestica iz sredine raspodjele i suboptimalnog ubrzavanja svih ostalih čestica. Usporedbom s CERN-ovim Linac4 akceleratorom pokazano je da se ovaj algoritam može koristiti i za simulaciju stvarnih akceleratora u određenoj mjeri, a ne samo za vrlo jednostavne modele.

Numeričko traženje optimalnih akceleratora je potrebno i za izgradnju stvarnih akceleratora jer se gotovo nijedan kompliciran problem u fizici ne može riješiti analitički. Nekakvim jednostavnim modifikacijama ovog algoritma mogli bi se dobiti još bolji rezultati za duljine cijevi i razmake među njima. Prva stvar koja se u tu svrhu može napraviti je promijeniti oblik električnog polja koji se pojavljuje između cijevi. Konstantno električno polje, koje se dobilo iz linearnog potencijala, je najjednostavnija moguća aproksimacija i u ovom algoritmu je bila korisna jer se mogao izvesti analitički izraz za provjeru. Za potpuno realnu simulaciju RF šupljina bilo bi potrebno krenuti od Maxwellovih jednadžbi i uvrstiti potrebne rubne uvjete, ali velik napredak u odnosu na ovu simulaciju postigao bi se aproksimiranjem stojnih elektromagnetskih valova koji nastanu u šupljinama sa sinusoidama kao što je prikazano na slici 3. Još jedan jako bitan faktor, koji je ovdje bio izostavljen radi jednostavnosti, je simuliranje kvadrupolnih magneta koji služe za fokusiranje snopa čestica. Gubitak ovako velikog broja čestica bio bi vrlo nepraktičan, stoga je nekakav mehanizam fokusiranja snopa nužan. Cijeli kod dostupan je na mom Github profilu², a algoritam se lako može poboljšati promjenom oblika električnog polja unutar jedne funkcije i dodavanjem proizvoljnih magnetskih polja koja su u ovom radu bila postavljena na 0.

²<https://github.com/rbarac1/Bachelor-s-Thesis>

7 Literatura

- [1] Thomson, J. J. *Cathode rays* (Friday evening meeting of the Royal Institutions, 30.4.1897.), *The Electrician*, 39, str. 104.
- [2] G. Ising: *Prinzip einer Methode zur Herstellung von Kanalstrahlen hoher Voltzahl*. In: *Arkiv för Matematik, Astronomi och Fysik*. Band 18, Nr. 30, 1924, S. 1–4.
- [3] Wideröe, R. (17 December 1928) "Über ein neues Prinzip zur Herstellung hoher Spannungen". *Archiv für Elektronik und Übertragungstechnik*. **21** (4): 387–406.
- [4] D. Alesini, [arXiv:2103.16500 [physics.acc-ph]]., URL: <https://arxiv.org/pdf/2103.16500.pdf>
- [5] Gerigk, Frank. (2011). "Cavity types". *CAS 2010 - CERN Accelerator School: RF for Accelerators, Proceedings.*, URL: https://www.researchgate.net/publication/51957233_Cavity_types (18.9.2022.)
- [6] Møller, C. (1955) [1952]. *The theory of relativity*. Oxford Clarendon Press., str. 74-75, URL: <https://archive.org/details/theoryofrelativi029229mbp/page/n91/mode/2up> (18.9.2022.)
- [7] Rice University. *University Physics - Volume 3* URL: <https://openstax.org/books/university-physics-volume-3/pages/5-9-relativistic-energy> (23.9.2022.)
- [8] Jessica Scarpati. "What is radio frequency (RF, rf)?". *SearchNetworking.*, URL: <https://www.techtarget.com/searchnetworking/definition/radio-frequency> (18.9.2022.)
- [9] CERN - Linear accelerator 4, URL: <https://home.cern/science/accelerators/linear-accelerator-4> (21.9.2022.)