

SVEUČILIŠTE U ZAGREBU  
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

Luka Greblo

# Simulacija ionskog vjetra temeljem Lattice-Boltzmannove metode

Zagreb, 2024.

*Ovaj rad je izrađen u Labroatoriju za robotiku i inteligentne sustave upravljanja, na Zavod za automatiku i računalno inženjerstvo pod vodstvom prof.dr.sc. Stjepana Bogdana i predan na natječaj za dodjelu Rektorove nagrade u akademskoj godini 2023./2024.*

# SADRŽAJ

<b>1. Uvod</b>	<b>1</b>
<b>2. Opći i specifični ciljevi rada</b>	<b>2</b>
<b>3. Koronalni izboj</b>	<b>3</b>
3.1. Matematički model . . . . .	3
3.1.1. Modeliranje gibanja iona . . . . .	6
3.2. Trenutne spoznaje . . . . .	8
<b>4. Metode rješavanja i simulacijska okruženja</b>	<b>12</b>
4.1. Metoda konačnih volumena (FVM) . . . . .	12
4.1.1. OpenFOAM implementacija . . . . .	13
4.2. Metoda konačnih elemenata (FEM) . . . . .	15
4.2.1. Elmer implementacija . . . . .	16
4.3. Lattice-Boltzmann metoda (LBM) . . . . .	17
4.3.1. OpenLB implementacija . . . . .	18
4.4. Usporedba LBM-a, FVM-a i FEM-a . . . . .	20
<b>5. Lattice-Boltzmanova metoda</b>	<b>22</b>
5.1. Razvoj Lattice-Boltzmann metode . . . . .	22
5.2. Osnovne definicije LBM-a . . . . .	22
5.2.1. Distribucijska funkcija i osnovne jednačbe . . . . .	22
5.3. Zakoni očuvanja i derivacija transportnih jednačbi . . . . .	24
5.4. Dimenzionalna Analiza . . . . .	25
5.4.1. Karakteristične Duljine, Vremena i Brzine . . . . .	25
5.4.2. Definiranje Bezdimezionalnih veličina . . . . .	25
5.4.3. Bezdimezionalni Brojevi . . . . .	26
5.4.4. Primjena u Lattice-Boltzmann metodi (LBM) . . . . .	26
5.4.5. Implementacija skaliranja u OpenLB-u . . . . .	27
5.5. Metode Diskretizacije . . . . .	27

5.5.1. D2Q9 Metoda . . . . .	28
<b>6. Trenutni načini simulacije</b>	<b>30</b>
6.1. Numerička studija elektroaerodinamičke sile i struje u sustavima s emiterom i kolektorom . . . . .	30
6.1.1. Elektroaerodinamičke sile . . . . .	30
6.1.2. Ionski vjetar i parametri sustava . . . . .	31
6.2. Uloga nelinearnih sila u elektrostatičkom propeleru pokretanom atmosferskim DC koronalnim pražnjenjima . . . . .	32
6.2.1. Elektrodinamičke sile . . . . .	32
6.2.2. Ionski vjetar i parametri sustava . . . . .	33
6.2.3. Numerička simulacija i eksperimentalni rezultati . . . . .	34
<b>7. Simulacija</b>	<b>36</b>
7.1. Postavljanje domene . . . . .	36
7.2. Električna svojstva unutar domene . . . . .	37
7.2.1. Izračun električnog potencijala i polja . . . . .	37
7.2.2. Primjena rezultata za spregu s dinamikom fluida . . . . .	41
7.2.3. Određivanje smjera sile i iznosa . . . . .	43
7.2.4. Postavljanje simulacije i uvjeta u domeni . . . . .	46
7.2.5. Pokretanje simulacije . . . . .	51
<b>8. Rezultati</b>	<b>52</b>
<b>9. Zaključak</b>	<b>60</b>
<b>Literatura</b>	<b>61</b>



# 1. Uvod

Čvrsta elektrostatička pogonska tehnologija dovodi do revolucije u područje svemirske propulzije i način na koji se ostvaruje kretanje u vakuumu. Za razliku od tradicionalnih kemijskih pogonskih sustava, elektrostatički pogon koristi električki nabijene čestice za generiranje pogonske sile, što se pokazalo efikasnijim prema dosadašnjim istraživanjima. U ovom radu bit će prikazani simulacijski scenariji koji opisuju ove sustave u atmosferskim uvjetima. Jedan od sve popularnijih oblika čvrstih elektrostatičkih pogona su ionski potisnici. Ionski potisnici uspješno su korišteni u vakuumskim uvjetima svemira uz primjenu inertnih plinova, no nedavni napredci u tehnologiji omogućili su njihovu primjenu i u atmosferi. U ovom radu bit će opisane osnove rada ionskih potisnika te njihove potencijalne primjene u atmosferskom okruženju, kao i već postignuta tehnička dostignuća. Iako je ionska propulzija često povezana sa svemirskim istraživanjima, ona pruža i značajne mogućnosti za poboljšanje performansi letjelica u atmosferskim uvjetima. Jedna od ključnih komponenti ovog tehnološkog pristupa je koronski izboj, koji ima središnju ulogu u stvaranju potiska bez pokretnih dijelova. Za istraživanje svojstava ovog tipa pogona nužno je razviti digitalnog blizanca (engl. *digital twin*), koji omogućuje provođenje simulacija radi optimizacije, bržeg razvoja i dubljeg razumijevanja efekata koje uzrokuje interakcija ionskog vjetra i tokova zraka.

## 2. Opći i specifični ciljevi rada

Primarni cilj ovog rada je istražiti učinkovitost i preciznost metode Lattice-Boltzmann (LBM) u simulaciji ionskog vjetra, s posebnim naglaskom na aproksimaciju sile koja djeluje na fluid. Ionski vjetar, fenomen koji se javlja pod utjecajem električnog polja na ione u fluidu, predstavlja kompleksan problem spregnutih fizikalnih procesa. Korištenje LBM metode za ovu svrhu predstavlja inovativan pristup koji omogućuje izravnu simulaciju fluida na mikroskopskoj razini, uz diskretizaciju prostora i vremena. Ova metoda ima potencijal pružiti prednosti u odnosu na klasične metode konačnih volumena (FVM), koje se tradicionalno koriste u softverskim paketima poput OpenFOAM-a. Usporedba rezultata dobivenih LBM metodom s onima dobivenim FVM metodama, s fokusom na točnost aproksimacije sile koja djeluje na fluid, odnosno na preciznost simulacije. Evaluacija računalnih resursa potrebnih za izvođenje simulacija pomoću obje metode, uključujući vrijeme potrebno za konvergenciju i stabilnost numeričkih rješenja, odnosno numeričku efikasnost metoda. Provjera konzistentnosti rezultata s fizičkim očekivanjima i eksperimentalnim podacima, uključujući analizu dinamike fluida pod utjecajem električnog polja. Identifikacija specifičnih prednosti LBM pristupa u simulaciji ionskog vjetra, kao i potencijalnih ograničenja ili izazova u primjeni ove metode. Ovaj rad doprinosi postojećem znanju na nekoliko načina.

Prvo, pruža dubinsku analizu sposobnosti LBM metode za precizno modeliranje kompleksnih spregnutih procesa u ionskom vjetru.

Drugo, usporedbom s FVM metodama, rad pruža uvid u potencijalne prednosti i nedostatke svake metode, nudeći smjernice za budući razvoj i optimizaciju numeričkih alata za simulaciju elektrohodinamičkih fenomena.

Konačno, rezultati ovog istraživanja mogu imati šire implikacije za primjene gdje su precizne simulacije fluida i elektrostatičkih sila ključne, poput dizajna naprednih propulzijskih sustava.

Tako rad ne doprinosi samo teorijskom razumijevanju, već i praktičnim primjenama u inženjerskim i znanstvenim disciplinama.

## 3. Koronalni izboj

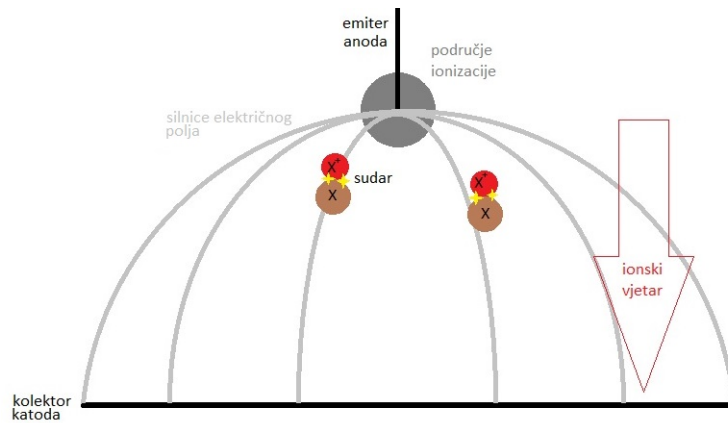
### 3.1. Matematički model

Ionska propulzija zasnovana je na ionizaciji atoma sredstva (plina) koja se odvija pod utjecajem visokog napona između dviju elektroda odmaknutih u sredstvu koje se ionizira. Plin u kojem su elektroni odvojeni od iona te se slobodno gibaju sredstvom, nalazi se u stanju koje se naziva plazma. Pri ionizaciji zraka razlikuje se nekoliko regija plina u blizini elektroda te se svojstva naglo mijenjaju na granicama. Regije oko elektroda su područja koronalnog izboja i područje električnog luka koji povezuje elektrode uz napone dovoljno visoke za probijanje otpora zraka i prelaska u vodljivi režim fluida. Prije prelaska u vodljivi režim iskrenja nakon napona proboja zraka, za vrijeme koronalnog izboja ioni putuju između elektroda te uslijed interakcija s neioniziranim dijelom sredstva stvara gibanje. Ovaj efekt naziva se ionski vjetar (engl. *ionic wind*), koji je generiran ionizacijom plina (stvaranjem plazme). Izvedba ionskog propulzora sastoji se od anode i katode koje su odvojene specifičnim razmakom ovisno o sredstvu u kojem se nalaze, naponu koji je doveden te geometrijskim svojstvima ovisno o ciljanom vladanju. Razlikuju se dvije vrste propulzora na temelju raspodjele elektroda, odnosno radi li se o pozitivnom ili negativnom koronalnom izboju.

Uzevši u obzir standardni sastav zraka koji glasi:

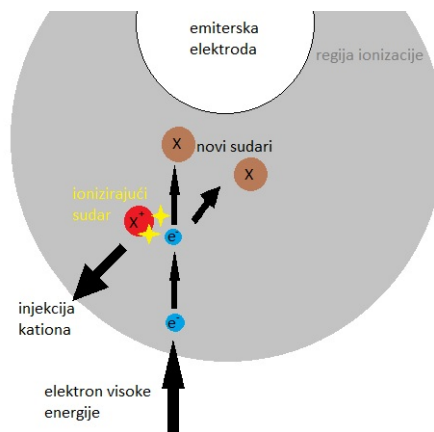
- 78.084% dušik,
- 20.946% kisik,
- 0.934% argon,
- 0.036% ugljikov dioksid,

moćno je pojednostaviti model kao ionizaciju samo dušika ( $N_2$ ) i kisika ( $O_2$ ), što čini ukupno 99.03% ukupnog fluida. Geometrija sustava u kojem je emiter točkasti izvor visokog napona postavljena iznad kolektora prikazan na slici 3.1 oznakom  $X$  uz to je prikazan atom kisika ili dušika zbog preglednosti. Dominantna reakcija koja se događa pri sudarima je  $e^- + X \longleftrightarrow 2e^- + X^+$ .



**Slika 3.1:** Shema sudara iona pri visokom naponu i generiranje ionskog vjetra

Uz direktnu ionizaciju u području ionizacije oko elektrode dolazi i do lavine multiplikacije ionizacije pri kojoj se slobodni elektroni nasumično sudaraju te uslijed velike količine energije koju nose imaju mogućnost stvaranja novih iona uslijed sudara. Ovaj fenomen poznat je kao Townsendova lavina, a princip je prikazan na slici 3.2. Ovakav izboj dovodi do velikog povećanja u vodljivosti materijala te time čini ključni korak u stvaranju iskrenja i proboja zraka pri naponu većem ili jednakom specifičnom proboju zraka u danim uvjetima.



**Slika 3.2:** Nastanak Townsendove lavine

Kako bi bilo moguće ispravno matematički modelirati sustav koronalnog izboja i povećanih efekata potrebni su modeli više fenomena. Neki od značajnijih efekata su:

- električna struja uslijed gibanja iona
- toplinska energija koja utječe na gibanje i brzine molekula
- Jouleovi efekti koji utječu na svojstva fluida

Uzimajući otvorenu domenu (neograničeni prostor)  $\Omega$  u smjerovima u kojima jednu stranicu ne zatvara kolektorska elektroda prema slici 3.1,  $\Omega$  predstavlja područje prostora koje je ispunjeno pretežno neutralnim fluidom i pozitivno nabijenim ionima. Područje ionizacije

oko emitera je zanemarivo s obzirom na ukupnu putanju između elektroda te je na slici 3.1 ona ilustrativno prikazana preuveličano.

Formulacija Poissonove jednadžbe za unipolarne ione u fluidu glasi:

$$\nabla \cdot (\epsilon \vec{E}) = -\nabla \cdot (\epsilon \nabla \phi) = qN_p \quad (3.1)$$

Sprega s kontinuitetom električne struje definira se jednadžbom 3.2:

$$\frac{\partial qN_p}{\partial t} + \nabla \cdot \vec{j} = 0 \quad (3.2)$$

U gore navedenim jednadžbama  $\epsilon$  je električna permitivnost zraka,  $\vec{E}$  vektor električnog polja,  $\phi$  električni potencijal,  $N_p$  broj protona te je  $q$  jedinični naboj. Sprega je u parcijalnim diferencijalnim jednadžbama povezana kroz gustoću struje  $\vec{j}$ , koja se povezuje s tri različita efekta: zanošenje uslijed djelovanja električnog polja, advekcija i difuzija, kao što je prikazano u jednadžbi 3.3:

$$\vec{j} = qN_p \cdot \mu \vec{E} + qN_p \vec{v} - qD \nabla N_p \quad (3.3)$$

Koeficijent  $\mu$  predstavlja mobilnost iona u fluidu te će biti dodatno prikazan u nastavku rada kao i njegova uloga u analitičkim izrazima nužnim za uvedene aproksimacije, dok je  $\vec{v}$  ukupna brzina fluida. Konstantu  $D$  nazivamo difuzijskom konstantom, koja je povezana s mobilnošću i temperaturom  $T$  prikazanom u Einsteinovoj jednadžbi, uz  $k_B$  kao Boltzmannovu konstantu:

$$D = \mu k_B T q^{-1} \quad (3.4)$$

Aerodinamički efekti nekompresibilnog Newtonskog fluida opisani su Navier-Stokesovim jednadžbama 3.5, koje predstavljaju očuvanje mase i količine gibanja:

$$\begin{aligned} \frac{\partial \vec{v}}{\partial t} + (\vec{v} \cdot \nabla) \vec{v} &= \nu \nabla^2 \vec{v} - \nabla \tilde{p} + \frac{\vec{F}_{\text{EDH}} + \vec{F}_B}{\rho} \\ \nabla \cdot \vec{v} &= 0 \end{aligned} \quad (3.5)$$

gdje je  $\nu$  kinematički viskozitet,  $\tilde{p}$  modificirani (efektivni) tlak u polju,  $\vec{F}_{\text{EDH}}$  elektromagnetske sile koje čine Lorentzovu silu u režimu elektrohidrodinamičkih efekata te  $\vec{F}_B$  sila zanošenja koja djeluje na fluid u prisutnosti magnetskog polja.

Prebacivanjem pažnje na zadnji član jednadžbe 3.5 sastoji se od sume sila koje su elektrohidrodinamička  $F_{\text{EDH}}$  3.18 i uzgona  $F_B$ . Pošto se radi o jednom toku uniformnog fluida (zraka) i ograničenog temperaturnog gradijenta sila koja proizlazi iz elektrohidrodinamike i veličine samog postava moguće je uvesti Boussinesqovu aproksimaciju oko referentne temperature  $T_{\text{ref}}$ :

$$\vec{F}_B = \vec{g}[\rho(T) - \rho] = \vec{g}[\rho \beta_{\text{exp}}(T_{\text{ref}})(T - T_{\text{ref}})] \quad (3.6)$$

gdje su  $\beta_{\text{exp}}$  termalni koeficijent proširenja i  $\rho(T)$ , varijabilna gustoća, ovisna o temperaturi oko točke linearizacije (temperature i gustoće). Zaključno za termodinamiku vrijedi:

$$\frac{\partial T}{\partial t} + \vec{v} \cdot \nabla T - \frac{k}{\rho C_V} \nabla^2 T = \frac{\dot{Q}}{\rho C_V} \quad (3.7)$$

gdje je  $k$  toplinska vodljivost, a  $C_V$  specifični kapacitet mase. Toplinsku snagu koja se razvija,  $\dot{Q}$ , je moguće raspisati kao Jouleovo zagrijavanje uslijed djelovanja  $\vec{j}$  i mehaničke snage razvijene uslijed gibanja iona  $F_{\text{EDH}}$ :

$$\dot{Q} = \vec{j} \cdot \vec{E} - \vec{v} \cdot F_{\text{EDH}} = (qN_P \mu \vec{E} - qD \nabla N_p) \cdot \vec{E} \quad (3.8)$$

### 3.1.1. Modeliranje gibanja iona

Kako bi se pokrenulo koronalno pražnjenje, pad napona između emitera i kolektora mora premašiti određeni prag vrijednost koji označavamo s  $V_0$ , dok je odgovarajuća jačina električnog polja na emiteru označena s  $E_0$ . Opće prihvaćena Kaptsovljeva hipoteza (Kaptsov, 1947) kaže da slobodni naboj, emitiran koronalnim pražnjenjem pri naponima višim od  $V_0$  rezultira "zadržavanjem" jačine električnog polja emiteru na vrijednosti  $E_0$ . Iako  $V_0$  jako ovisi o cjelokupnoj geometriji uređaja, eksperimentalni dokazi pokazuju da je  $E_0$  strogo povezan s polumjerom zakrivljenosti kolektorske elektrode Peek (1929). Na mikroskali, koronalno pražnjenje se generira udarnom ionizacijom molekula plina i lavinskom multiplikacijom elektrona. Prema lavinskom modelu koji je prvi razvio Townsend i objavio u knjigama: Townsend i Edward (1910) i Townsend (1915), kationi se generiraju u području koje karakterizira skup točaka  $\vec{x} \in \Omega$  takav da:

$$\gamma_T \exp \left( \int_{L(\vec{x})} \alpha_T(\vec{r}) \cdot d\vec{r} \right) \geq 1 \quad (3.9)$$

Parametri  $\gamma_T$  i  $\alpha_T$  ovise o primijenjenom električnom polju, tlaku i kemijskom sastavu plina i elektroda, dok je  $L(\vec{x})$  putanja negativno nabijenih čestica koje kreću s emitera i putuju do točke  $\vec{x}$  zbog sile električnog polja. Iako nije praktično važna kada prostorni naboj nije zanemariiv, relacija 3.9 daje grubu procjenu debljine ionizacijskog područja gdje se plin može smatrati plazmom. Ta debljina uvelike ovisi o geometriji, tlaku fluida i električnom polju. U režimu koronalnog izboja ona je toliko mala u usporedbi s duljinom neutralnog područja fluida da ima smisla koristiti objedinjeni model za ionizacijsko područje i predstaviti ga kao dio površine emitera. U ovoj aproksimaciji, jedini nositelji naboja unutar područja fluida su kationi kako je i opisno u radu Peek (1929).

Uzevši opći oblik modeliranja stope ubrizgavanja kationa u regiju glavnog fluida. S ciljem opisa sustava potrebno je odrediti rubni uvjet odnosno u ovom slučaju jednadžbu rubnog uvjeta Robinovog tipa kao u jednadžbi 3.10 u kojoj  $\Gamma_A$  označava površinu emitera:

$$\alpha N_p|_{\Gamma_A} + \beta \partial_{\mathbf{n}} N_p|_{\Gamma_A} = \kappa \quad (3.10)$$

gdje je  $\partial_{\vec{n}}N_p = \nabla N_p \cdot \vec{n}$  komponenta gradijenta gustoće iona normalna na  $\Gamma_A$ . Uvjet 3.10 općenito će biti nelinearan jer dopušta koeficijente  $\alpha$ ,  $\beta$  i  $\kappa$  koji lokalno ovise o normalnoj komponenti  $E_n = \vec{n} \cdot \vec{E}|_{\Gamma_A}$  i o gustoći iona  $N_p|_{\Gamma_A}$ . Najčešći pristup korišten u numeričkim studijama pozitivnog korona pražnjenja koji se pojavljuje u Mazumder i Lai (2011) Mazumder et al. (2011) sastoji se u postavljanju struje  $i_m$  tako da na emiteru bude jednaka eksperimentalno izmjerenoj vrijednosti  $i_m$ . To vodi do sljedećeg izbora parametara u 3.10:

$$\begin{aligned}\alpha_1 N_p|_{\Gamma_A} + \beta_1 \partial_{\mathbf{n}} N_p|_{\Gamma_A} &= \kappa_1, \\ \alpha_1 &= -q\mu E_n, \\ \beta_1 &= qD, \\ \kappa_1 &= \frac{i_m}{s}\end{aligned}\tag{3.11}$$

Vidljivo je da se prema 3.11 model temelji na dodatnoj pretpostavci da je komponenta gustoće ionske struje kao  $j_n = \vec{n} \cdot \vec{j}|_{\Gamma_A}$  normala na kontakt jednoliko raspoređena duž  $\Gamma_A$  (što predstavlja uniformnu razdiobu). Ova posljednja pretpostavka, zajedno s činjenicom da je potrebno poznavanje izmjerene vrijednosti struje za svaku vrijednost primijenjenog napona, značajno ograničava sposobnost simulacija i izračuna koji su temeljeni na jednadžbama 3.11 što onemogućuje pružanje korisnih informacija o utjecaju geometrije kontakta emitera na performanse uređaja. Postoji modifikacija za modeliranje brzine kojom se kationi ubrizgavaju u područje fluida. Kako bi bilo moguće pojednostaviti izračun iz kojeg je moguće izvesti jednadžbu prema 3.23 koja je korištena u Sigmond (1982), a predložena prema Thomas Christen (2005)

$$j_n = wN_p - j_{\text{sat}}H(E_n - E_0)\tag{3.12}$$

gdje  $H(x)$  označava Heavisideova funkciju skoka. Parametri koji se pojavljuju u 3.12 su maksimalna dozvoljena gustoća struje  $j_{\text{sat}}$ , prag električnog polja  $E_0$  za stvaranje ionizacije, te proporcionalna konstanta  $w$  (dimenzije brzine pomnožene s električnim nabojem) između struje povratnog raspršenja i količine iona nakupljenih u regiji prostornog naboja na emiteru. Ovaj model je moguće predstaviti kao struju kontroliranu prostornim nabojem. Korištenjem 3.12 za određivanje koeficijenata općeg izraza 3.10 dolazimo do

$$\begin{aligned}\alpha_2 N_p|_{\Gamma_A} + \beta^2 \partial_{\mathbf{n}} N_p|_{\Gamma_A} &= \kappa_2, \\ \alpha_2 &= w - q\mu E_n, \\ \beta_2 &= qD, \\ \kappa_2 &= j_{\text{sat}}H(E_n - E_0).\end{aligned}\tag{3.13}$$

Iako ovaj model ne zahtijeva prethodno znanje o gustoći struje, čime se naizgled rješava glavni problem modela 3.12, kvaliteta njegovih predikcija kritično ovisi o ispravnom izboru parametara  $j_{\text{sat}}$  i  $w$ , te se čini da je, u nekim relevantnim praktičnim situacijama, prilično loša

ako se tim parametrima daju vrijednosti neovisne o pristranosti. Poboljšanjem je moguće uvesti još daljnje promjene modela 3.12 kod  $\kappa_2$  u  $\kappa_3$  tako da se model definira kao:

$$\begin{aligned}\alpha_3 N_p|_{\Gamma_A} + \beta^2 \partial_{\mathbf{n}} N_p|_{\Gamma_A} &= \kappa_3^2, \\ \alpha_3 &= w - q\mu E_0, \\ \beta_3 &= qD, \\ \kappa_3 &= q\mu E_n N_p|_{\Gamma_A}\end{aligned}\tag{3.14}$$

Pristup prikazan u modelu opisanom jednadžbom 3.14 ima samo jedan parametar kojega je relativno lako odrediti, čak i aproksimativno, a to je  $E_0$ , no pod cijenu nelinearnosti na granici kroz  $N_p|_{\Gamma_A}$ , što u stacionarnom stanju polja kakva su u ovome radu razmatrane omogućuje, jednostavnu linearizaciju u ustaljenom stanju.

## 3.2. Trenutne spoznaje

Koronalni izboj događa se između elektrode s oštro zakrivljenom površinom, poput oštrice ili tankog vrha žice i površinom veće, ali oblikom mnogo manje zakrivljene površine ili čak ravnine. Vrh ili žica nazivaju se emitorskom elektrodom i mogu biti pozitivno ili negativno nabijeni u odnosu na suprotnu elektrodu. Druga elektroda naziva se kolektorskom elektrodom i suprotnog je polariteta ili je uzemljena u odnosu na emitorsku elektrodu. Koronalni izboj oblik je pražnjenja u kojem se, osim u malom području oko emitera, struja prenosi isključivo pomoću iona. Napon između emitorske i kolektorske elektrode postavlja se u red veličine kV u zraku pod atmosferskim tlakom te taj napon može biti pozitivan ili negativan, ovisno o naboju na emitorskoj elektrodi. Uslijed veoma malih dimenzija emitera, stvara se vrlo jako lokalno električno polje u blizini oštrog ruba. Red veličine električnog polja u zraku u bliskoj okolini emitorske elektrode iznosi  $10^7$  V/m. Tolika jakost električnog polja dovodi do elektronske emisije, što rezultira time da, ako je emiter pozitivno nabijen, elektroni će se vratiti emiteru, ali će pri tom proći kroz sudare s molekulama zraka, stvarajući pozitivno nabijene ione. Time će tako nastali ioni biti odbačeni od emitera i privučeni prema kolektoru, pri čemu će nasumično dolaziti do sudara s molekulama zraka, dajući im moment i time generirajući ionski vjetar, koji je osnova za stvaranje potiska. Područje u kojem se ioni stvaraju vrlo je malo u usporedbi s udaljenosti između emitera i kolektora, tako da većinom područja pražnjenja dominira ionska struja. Debljina  $\delta$  (mm) ionizacijskog područja može se izračunati prema radu E.A. i P.S. (1967)

$$\delta = \sqrt{a}\tag{3.15}$$

gdje je  $a$  (mm) radijus vrha emitera iz kojeg se javlja koronalni izboj. Ako se primijenjeni napon previše poveća i približi naponu proboja zraka, dolazi do iskrenja za tu duljinu zračnog



raspora, a svjetlost korone prelazi u iskru. Christen i Moller (E.A. i P.S., 1967) pokazali su da potisak koji proizvodi ionski vjetar u zraku,  $T$  (N), može se zapisati kao:

$$T = \frac{I \cdot d}{\mu} \quad (3.16)$$

gdje je  $I$  struja u pražnjenju (A),  $d$  raspon između elektroda emitera i kolektora (m), a  $\mu$  ionska mobilnost ( $m^2/V \cdot s$ ). S obzirom na potisak izražen kao potisak po jedinici snage  $\theta$ , moguće ga je prikazati kao:

$$\theta = \frac{T}{IV} = \frac{d}{V} = \frac{1}{E\mu} \quad (3.17)$$

gdje je  $V$  primijenjeni napon, a  $E$  prosječno električno polje, definirano kao  $E = \frac{V}{d}$ . Problem korištenja ionske propulzije kao pogona vidljiv je u jednadžbi 3.17. Budući da bi se postigao visoki omjer potiska prema snazi, električno polje mora biti nisko, što implicira nisku struju, a time i nizak potisak prema jednadžbi 3.15. Alternativa je imati nisku mobilnost. Mobilnost zraka poznata je i ovisi o vlažnosti Tyndall i Grindley (1926), ali varira samo od vrijednosti  $2.15 \times 10^{-4} m^2/Vs$  za suhi zrak do  $1.6 \times 10^{-4}$  za zasićeni zrak. Iz eksperimenata E.A. i P.S. (1967) zaključeno je da učinkovitost pogona u atmosferskim uvjetima iznosi oko 1%, što je premalo za bilo kakvu praktičnu primjenu. Njihov maksimalno izmjereni potisak bio je oko 0.45 N, s površinom protoka otprilike 0.1 m<sup>2</sup>, a omjer potiska prema snazi bio je 21 N/kW. Jedina geometrijska varijabla u eksperimentu bila je udaljenost između elektroda. Počevši od jednadžbe 3.15, može se izvesti vrlo jednostavno pretpostavljajući, kao što je to učinjeno u radu E.A. i P.S. (1967), da je električno polje uniformno u glavnom dijelu koronalnog pražnjenja te će ta pretpostavka biti korištena i u aproksimaciji polja na kojem djeluje sila u ovom radu, s primijenjenim naponom  $V$  (volti). Budući da više nema iona nakon što su ioni napustili područje visokog polja oko vrha, umnožak gustoće iona,  $n_i$  i presjeka pražnjenja  $A$  bit će konstantan. Svaki ion doživjet će elektrostatičku silu,  $T_i$ , definiranu kao:

$$T_i = eE \quad (3.18)$$

gdje je  $e$  jedinični naboj, a naravno postoje iznosom jednake i smjerom suprotne sile na elektrodama. Ukupna sila svih iona na elektrodama tada je:

$$T = NT_i = NeE \quad (3.19)$$

gdje je  $N$ , ukupan broj iona u pražnjenju, jednostavno  $N = n_i A d$ . Struja pražnjenja dana je:

$$I = n_i e v_i A \quad (3.20)$$

gdje je  $v_i$  prosječna brzina iona

$$v_i = \mu E \quad (3.21)$$

Slijedi da je ukupna sila na elektrodama:

$$\begin{aligned}
 T &= NeE \\
 &= n_i A d e E \\
 &= \frac{(n_i e v_i A) d E}{\mu E} \\
 &= \frac{I d}{\mu}
 \end{aligned} \tag{3.22}$$

što se poklapa s jednadžbom 3.15. Ovo pokazuje da je potisna sila povezana s ionskim vjetrom jednostavno elektrostatičko odbijanje emitera, odnosno privlačenje kolektora na oblak iona u pražnjenju. Drugi način za određivanje struje u mirnom zraku je korištenjem Townsendovog zakona, koji glasi:

$$I = C \cdot V \cdot (V - V_0) \tag{3.23}$$

Gdje je  $C$  konstanta koja opisuje probojna svojstva fluida s mjernom jedinicom  $A^3 s^6 / (m^4 kg^2)$ ,  $V_0$  napon pri kojem počinje utjecaj ionizacije te  $V$  napon koji je doveden na razmak među elektrodama. S ciljem numeričkog određivanja struje, potrebno je provesti integraciju gustoće naboja kroz cijelo područje ionizacije, no moguće je uvesti pojednostavljenje koje pretpostavlja uniformnu gustoću naboja u cijelom području ionizacije kao gustoću naboja danom s  $\rho_c$  u području površine  $A$  prema radu (Sigmond, 1982), te je brzina iona označena kao  $v_i$  sukladno jednadžbi 3.20 gdje je gustoća zamijenjena pojedinačnim česticama.

$$I = \rho_c \cdot v_i \cdot A \tag{3.24}$$

Nadalje, nužno je dodati i brzinu okolnog fluida unutar kojeg se kreće sam potisnik u slučaju da takvo strujanje postoji te je tada ukupna brzina iona  $v_i$  sastavljena kao zbroj brzina okolnog fluida  $v_f$  i brzine čestice u polju  $E$  unutar fluida na temelju mobilnosti  $\mu$ :

$$v_i = v_f + \mu E \tag{3.25}$$

Brzina iona u gore navedenom izvodu ovisila je o ionskoj mobilnosti, koja se određuje sudarima iona sa zrakom kroz koji putuju. Sam ionski vjetar stvara se pri svakom sudaru iona s molekulom zraka, što ometa kretanje iona prema kolektoru i ubrzava zrak prema kolektoru. Jednostavan model sudara ilustrirat će to.

Ioni su u uniformnom polju, pa će se ubrzavati pod konstantnim ubrzanjem  $a$ , dano kao:

$$a = \frac{eE}{m_i} \tag{3.26}$$

gdje je  $m_i$  masa iona. Na putu duljine  $\lambda$ , tj. srednje slobodne staze iona u zraku, ion će se sudariti s molekulom zraka. Ako se dodatno pretpostavi da će se u ovom sudaru ion dovesti

do zaustavljanja, a molekula zraka dobiti svu količinu gibanja koju je ion imao prije sudara, tada će ion prevaliti udaljenost  $\lambda$  u vremenu  $t$  tako da:

$$\lambda = at^2 \quad (3.27)$$

iz čega proizlazi prosječna brzina iona kao:

$$\begin{aligned} v_i &= \frac{\lambda}{t} \\ &= \frac{at}{2} \\ &= \frac{eE}{2m_i} \cdot \sqrt{\frac{2m_i\lambda}{eE}} \\ &= E \sqrt{\frac{e\lambda}{2m_iE}} \end{aligned} \quad (3.28)$$

no  $v_i/E$  je mobilnost  $\mu$ , iz čega slijedi:

$$\mu = \sqrt{\frac{e\lambda}{2m_iE}} \quad (3.29)$$

Rezultat da je  $\mu$  proporcionalna s inverznim korijenom polja podudara se s rezultatima za ione u zraku pod visokim električnim poljem prema navodu Sigmond (1982). Na udaljenosti puta  $d$  između emitera i kolektora, svaki ion će doživjeti  $d/\lambda$  sudara. Na svakom sudaru ion predaje zraku impuls  $2m_iv_i$ . To pretpostavlja sudare uglavnom s jednakom masom iona i molekulama zraka. Dakle, ukupni impuls koji se daje zraku, uzimajući u obzir vrijeme, može se tumačiti kao sila koja ubrzava zrak,  $F_{EDH}$ , što je:

$$F_{EDH} = n \cdot 2m_iv_i \cdot \frac{d}{\lambda} \quad (3.30)$$

pri čemu je  $n$  broj iona koji dolazi do kolektora u jedinici vremena (s), što je moguće prikazati kao promjenu naboja u vremenu po ionu  $I/e$ . Uvrštavanjem te činjenice u jednadžbu 3.28, kao i jednadžbe 3.30 za  $v_i$ , u jednadžbu koja uvodi fizikalnu konstantu mobilnosti  $\mu$  3.29, sila koja ubrzava zrak je:

$$F_{EDH} = I \cdot d \sqrt{\frac{2m_i \cdot E}{e\lambda}} \quad (3.31)$$

Kompleksno određivanje mobilnosti elektrona u zraku nije moguće izbjeći samim uvrštavanjem mobilnosti 3.29 na odgovarajući član u analitičke izraze, ali je moguće izbjeći korištenje električnog polja uvrštavanjem u jednadžbu 3.31, te je takav rezultat vidljiv kao jednadžba 3.32:

$$F_{EDH} = \frac{I \cdot d}{\mu} \quad (3.32)$$

Dakle, vidi se da je sila koja ubrzava zrak, tj. sila koju stvara ionski vjetar, ista kao potisak na elektrodama, koji je sam reakcija električnoj sili na ione.

## 4. Metode rješavanja i simulacijska okruženja

Simulacija dinamike fluida često je korišten inženjerski alat te je u posljednje vrijeme doživio pravu revoluciju. Broj radova povezanih s dinamikom fluida naglo je porastao s napretkom raspoložive računalne snage te su samim time i modeli postali precizniji, a time i njihovi rezultati reprezentativniji. Ovakav razvoj omogućio je veliku raznolikost u načinu izvedbe pojedinih alata i razvoju specifičnih svojstava koje bi omogućili modeliranje mnogih efekata vezanih za dinamiku fluida. Razvijeno je mnogo komercijalnih, ali i alata za slobodnu uporabu (*open source*) te bit će razmatrane samo pojedine implementacije ovisno o metodi rada. Rast interesa je jasno opisan u analizi tržišta od strane EMRa te je utvrđeno kako je na tom području očekivan rast od 11.7% do 2032 uz vrijednost tržišta od oko 6.3mlrd \$. Svi su alati koji se koriste bazirani na diskretiziranim domenama i numeričkim metodama za rješavanje Navier-Stokesovih jednadžbi ili nekom od pojednostavljenih rješenja. U osnovi načini rješavanja razlikuju se ovisno o izvedbi njihovih metoda diskretizacije domene i primijenjenih numeričkih metoda koje im odgovaraju. Glavne skupine su FEM (*finite element method*) programi i FVM (*finite volume method*) te Lattice-Boltzmannova metoda koja kombinira statističku mehaniku fluida s mikroskopskim kinetičkim zakonima. U nastavku će biti razmatrane navedene metode te na temelju te usporedbe bit će potanje objašnjeno zašto je odabrana Lattice-Boltzmannova metoda.

### 4.1. Metoda konačnih volumena (FVM)

Metoda konačnih volumena (FVM) je jedna od najkorištenijih numeričkih tehnika za rješavanje parcijalnih diferencijalnih jednadžbi, posebno u kontekstu dinamike fluida. Osnovna ideja FVM-a je podjela domene na male, diskretne volumene (kontrolne volumene) i primjena integralnih oblika diferencijalnih jednadžbi preko tih volumena.

## Teoretska osnova FVM-a

Metoda konačnih volumena temelji se na integralnim oblicima zakona očuvanja (masa, količina gibanja, energija). Računska domena dijeli se na konačne volumene, te se integralni oblici jednadžbi primjenjuju unutar svakog volumena. Ova metoda osigurava da su ključni fizikalni zakoni očuvanja implicitno zadovoljeni na diskretnoj razini, što je jedna od najvećih prednosti FVM-a.

Za potrebe dobivanja rješenja korištene su jednadžba kontinuiteta za nestlačive fluide i Navier-Stokesova jednadžba. Jednadžba kontinuiteta govori o konzervaciji mase te glasi:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \quad (4.1)$$

gdje je  $\rho$  gustoća fluida,  $\mathbf{u}$  brzina fluida, a  $t$  vrijeme. U nestlačivim tokovima,  $\rho$  predstavlja konstantnu vrijednost, pa jednadžba kontinuiteta postaje:

$$\nabla \cdot \mathbf{u} = 0 \quad (4.2)$$

Navier-Stokesova jednadžba za nestlačive fluide uključuje vanjske sile:

$$\frac{\partial(\rho \mathbf{u})}{\partial t} + \nabla \cdot (\rho \mathbf{u} \mathbf{u}) = -\nabla p + \nabla \cdot (\mu \nabla \mathbf{u}) + \mathbf{F} \quad (4.3)$$

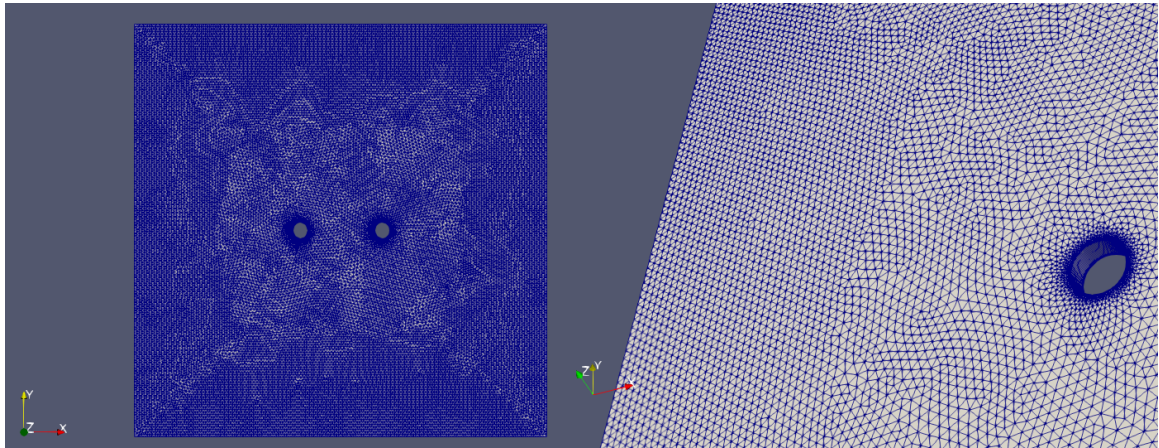
gdje je  $p$  tlak,  $\mu$  dinamička viskoznost, a  $\mathbf{F}$  vanjske sile (npr. gravitacija, elektromagnetske sile ili u kontekstu ovog rada EHD). Kako bi bilo moguće riješiti jednadžbe 4.2 i 4.3 potrebno je osigurati diskretne volumene na kojima se kontinuirana domena evaluira. Taj se postupak naziva diskretizacijom, a najčešće korištene sheme diskretizacije u FVM-u uključuju:

- Centralna razlika: koristi srednju vrijednost za diskretizaciju derivata, pogodna za laminarne tokove.
- Upwind shema: koristi vrijednosti iz uzvodnih ćelija, stabilnija je za turbulentne tokove.
- Shema drugog reda (QUICK, MUSCL): koristi polinomske aproksimacije za postizanje veće točnosti.

### 4.1.1. OpenFOAM implementacija

OpenFOAM koristi FVM za rješavanje Navier-Stokesovih jednadžbi, uključujući modele za Reynolds-averaged Navier-Stokes (RANS) i Large Eddy Simulation (LES). Detalji o implementaciji mogu se pronaći u dokumentaciji OpenFOAM-a Jasak i A. Jemcov (2007). Mreže koje diskretiziraju računsko područje mogu biti strukturirane ili nestrukturirane:

- Strukturirane mreže: imaju redovitu, pravokutnu topologiju, što olakšava implementaciju i smanjuje računalne troškove. Međutim, nisu pogodne za kompleksne geometrije.
- Nestrukturirane mreže: sastoje se od nepravilnih, trokutastih ili tetraedarskih elemenata, što omogućava fleksibilnost u modeliranju kompleksnih geometrija, ali povećava računalne troškove i složenost implementacije.



**Slika 4.1:** Shema podjele područja na konačne volumene u FVM.

### **Dobre strane FVM-a**

Jedna od ključnih prednosti FVM-a je konzervativna priroda metode. Integriranje parcijalnih diferencijalnih jednadžbi preko kontrolnih volumena osigurava očuvanje masa, količine gibanja i energije. Ovo je posebno važno u simulacijama dinamike fluida gdje su takvi zakoni očuvanja fundamentalni za točnost i stabilnost rješenja (H. K. Versteeg, 2007).

FVM je također vrlo fleksibilan u radu s nestrukturiranim mrežama. Ova fleksibilnost omogućava prilagodbu mreže složenim geometrijama koje se često javljaju u stvarnim inženjerskim problemima. Korištenje nestrukturiranih mreža omogućava detaljno modeliranje geometrijskih složenosti bez prekomjernog rafiniranja mreže (Moukalled i L. Mangani, 2016).

Primjena FVM-a u softverskom paketu OpenFOAM pruža dodatne prednosti. OpenFOAM je otvorenog koda, što omogućava korisnicima prilagodbu i proširenje postojećih modela prema specifičnim potrebama. OpenFOAM nudi širok spektar alata za simulaciju dinamike fluida, od osnovnih rješenja do naprednih turbulentnih modela i višefaznih tokova Jasak i A. Jemcov (2007).

## Loše strane FVM-a

Unatoč mnogim prednostima, FVM ima i svoje nedostatke. Jedan od glavnih izazova je složenost formulacije za visoko nestrukturirane mreže, što može dovesti do numeričkih nestabilnosti. Postizanje stabilnosti i točnosti često zahtijeva sofisticirane metode za generiranje mreže i aproksimaciju rubnih uvjeta (Blazek, 2001).

Također, dok je FVM odličan u očuvanju konzervativnih svojstava, može imati problema s točnošću kod rješavanja problema sa složenim granicama i prijelaznim stanjima kao što su sile na fluid korišten u sklopu ovog rada. Ove poteškoće često zahtijevaju dodatne mjere, poput rafiniranja mreže, što povećava računске troškove i vrijeme simulacije (J. H. Ferziger, 2002).

## 4.2. Metoda konačnih elemenata (FEM)

Metoda konačnih elemenata (FEM) je još jedna popularna tehnika za numeričko rješavanje parcijalnih diferencijalnih jednadžbi, koja se široko koristi u inženjerskim i fizikalnim problemima. FEM se temelji na podjeli problema na manje, konačne elemente unutar kojih se problem lokalno aproksimira.

### Teoretska osnova FEM-a

FEM se temelji na varijacijskom pristupu rješavanju diferencijalnih jednadžbi, gdje se originalna jednadžba transformira u slabi oblik koji je pogodan za diskretizaciju. Problem se zatim dijeli na konačne elemente, a aproksimacija se provodi korištenjem funkcija oblika koje su definirane unutar svakog elementa. Metoda konačnih volumena (FVM) koristi integralni oblik zakona očuvanja za diskretizaciju jednadžbi. Računsko područje se dijeli na konačne volumene te se integralni oblici jednadžbi primjenjuju unutar svakog volumena. Ova metoda osigurava očuvanje ključnih fizikalnih zakona na diskretnoj razini. Metoda konačnih elemenata (FEM) koristi varijacijski princip za diskretizaciju kontinuiranih polja. Problemi se postavljaju kao varijacijske forme koje se zatim diskretiziraju korištenjem konačnih elemenata.

U FEM-u, Navier-Stokesova jednadžba može se izraziti u slabom obliku:

$$\int_{\Omega} \frac{\partial(\rho \mathbf{u})}{\partial t} \cdot \mathbf{v} \, d\Omega + \int_{\Omega} (\rho \mathbf{u} \cdot \nabla) \mathbf{u} \cdot \mathbf{v} \, d\Omega = - \int_{\Omega} \nabla p \cdot \mathbf{v} \, d\Omega + \int_{\Omega} \mu \nabla \mathbf{u} \cdot \nabla \mathbf{v} \, d\Omega + \int_{\Omega} \mathbf{f} \cdot \mathbf{v} \, d\Omega \quad (4.4)$$

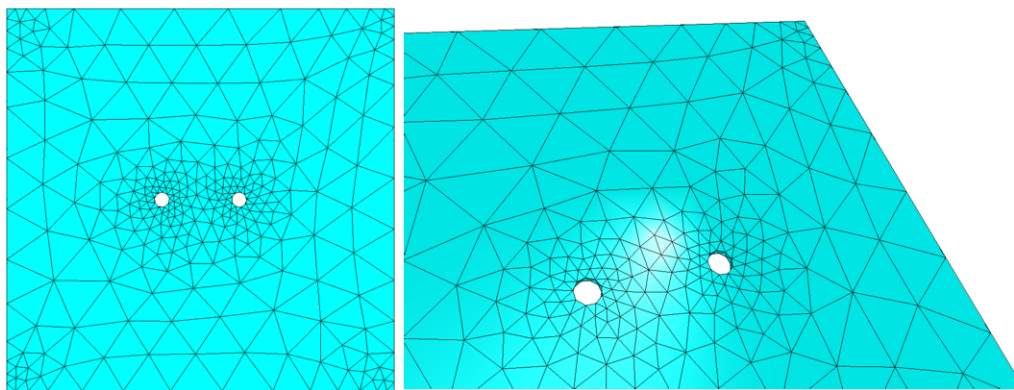
gdje je  $\mathbf{v}$  testna funkcija, a  $\Omega$  područje integracije.

Najčešće korištene sheme diskretizacije u FEM-u zasnovane su na izradi objekta mreže u kojima će se vršiti evaluacija jednadžbi uključuju:

- Linearni elementi: jednostavni za implementaciju, koriste se za osnovne simulacije.
- Kvadratni elementi: pružaju veću točnost, koriste se za složenije simulacije.
- Isoparametrični elementi: prilagodljivi za kompleksne geometrije.

### 4.2.1. Elmer implementacija

Elmer softver koristi FEM za rješavanje različitih fizikalnih problema, uključujući dinamiku fluida. Na slici 4.2 prikazana je diskretizacija domene za korištenje FEM metoda u programskom paketu Elmer koji omogućava fleksibilnu postavu problema koristeći razne rubne i početne uvjete (M. Malinen, 2008), no teško definira spregu, a ne ide ni u prilog činjenica da je program napisan u FORTRAN-u.



Slika 4.2: Shema konačnih elemenata korištenih u FEM simulacijama.

#### Dobre strane FEM-a

Jedna od ključnih prednosti FEM-a je njegova sposobnost preciznog modeliranja složenih geometrija i rubnih uvjeta kao što je vidljivo na slici 4.2. Ova metoda omogućava jednostavno definiranje različitih materijalnih svojstava unutar različitih elemenata, što je korisno u mnogim inženjerskim primjenama (O. C. Zienkiewicz, 2005).

FEM također nudi visoku fleksibilnost u izboru funkcija oblika, što omogućava visoku točnost u aproksimaciji rješenja. Ova fleksibilnost je posebno korisna u problemima gdje je potrebno precizno modeliranje gradijenata. Mogućnost korištenja višestrukih tipova elemenata i prilagodljivih mreža dodatno doprinosi točnosti simulacija (Bathe, 1996).

Softverski paket Elmer, koji koristi FEM, omogućava korisnicima napredne mogućnosti simulacije u dinamici fluida i termomehanike, s bogatim setom alata za prilagodbu i optimizaciju. Elmer je također otvorenog koda, što omogućava široku primjenu i prilagodbu specifičnim potrebama korisnika (M. Malinen, 2008), no kako je navedeno teško je mijenjati implementacije i dodavati nove sustave spregu između raznih fizikalnih efekata.



## Loše strane FEM-a

Unatoč svojim prednostima, FEM također ima nekoliko nedostataka. Jedan od glavnih problema je računaska složenost i potreba za velikim memorijskim kapacitetima, posebno kod trodimenzionalnih problema. Ovi zahtjevi mogu biti prepreka za primjenu FEM-a u vrlo velikim ili detaljnim simulacijama Hughes (1987).

Također, postavljanje mreže za FEM može biti složeno i zahtjeva veliko iskustvo i stručnost. Greške u mreži mogu značajno utjecati na točnost i stabilnost rješenja, što zahtijeva dodatnu pažnju pri definiranju i provjeri mreže (Cook et al., 2001).

## 4.3. Lattice-Boltzmann metoda (LBM)

Lattice Boltzmann metoda (LBM) je relativno novija numerička tehnika za simulaciju dinamike fluida koja se temelji na kinetičkoj teoriji i modelima diskretnih brzina. LBM ne rješava izravno Navier-Stokesove jednadžbe, već koristi pojednostavljeni model kinetičke teorije kako bi simulirala ponašanje fluida na mikroskopskoj razini.

### Teoretska osnova LBM-a

LBM se temelji na diskretizaciji prostora u rešetku (lattice) koja je poseban slučaj strukturirane mreže te na aproksimaciji distribucijske funkcije čestica unutar svakog čvora te rešetke. Evolucija distribucijske funkcije kroz vrijeme određuje makroskopska svojstva fluida kao što su brzina i tlak. Osnovne jednadžbe LBM-a su jednadžbe za evoluciju distribucijske funkcije, koje uključuju sudarne operatore koji modeliraju sudare između čestica (Succi, 2001). Lattice-Boltzmann metoda (LBM) temelji se na diskretizaciji prostora u rešetku i evoluciji distribucijske funkcije čestica kroz vrijeme. Ova metoda simulira ponašanje fluida koristeći mikroskopske modele kinetičke teorije. Osnovna Lattice Boltzmann jednadžba je:

$$f_i(\mathbf{x} + \mathbf{e}_i \Delta t, t + \Delta t) - f_i(\mathbf{x}, t) = -\frac{\Delta t}{\tau} [f_i(\mathbf{x}, t) - f_i^{\text{eq}}(\mathbf{x}, t)] \quad (4.5)$$

gdje je  $f_i$  distribucijska funkcija za česticu s brzinom  $\mathbf{e}_i$ ,  $\Delta t$  vremenski korak,  $\tau$  vrijeme relaksacije, a  $f_i^{\text{eq}}$  ekvilibrij distribucijska funkcija.

Makroskopske veličine kao što su gustoća  $\rho$  i brzina  $\mathbf{u}$  dobivaju se sumiranjem distribucijskih funkcija:

$$\rho(\mathbf{x}, t) = \sum_i f_i(\mathbf{x}, t) \quad (4.6)$$

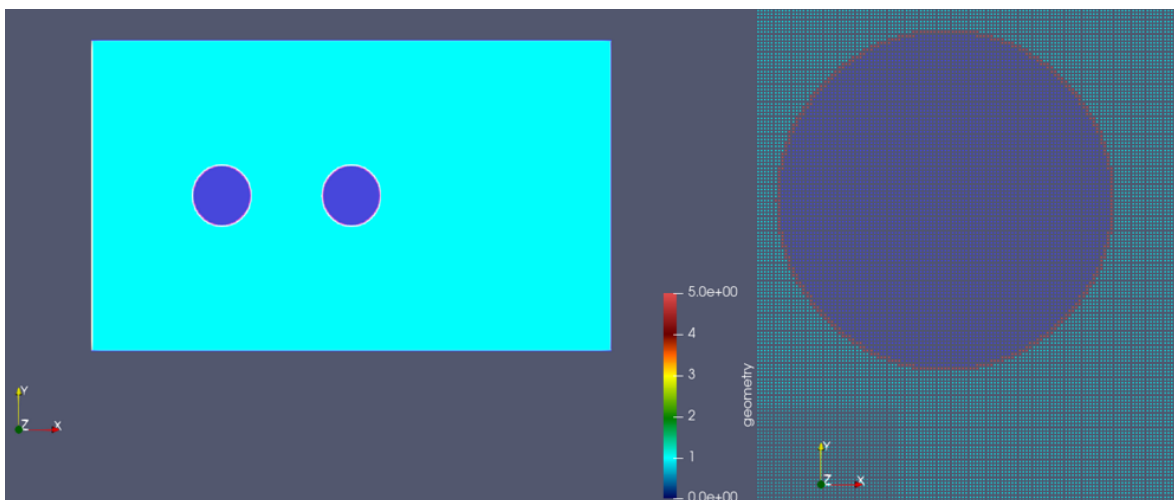
$$\rho \mathbf{u}(\mathbf{x}, t) = \sum_i f_i(\mathbf{x}, t) \mathbf{e}_i \quad (4.7)$$

Diskretizacija kao LBM ostvarena je po čvorovima unutar lattice koje je slična strukturiranoj mreži, no njeno je generiranje uvelike lakše kao što će biti opisano u nastavku. Najčešće korištene sheme diskretizacije u LBM-u uključuju:

- D2Q9 model: koristi 9 brzina u dvodimenzionalnoj rešetki.
- D3Q19 model: koristi 19 brzina u trodimenzionalnoj rešetki.
- Multiple-Relaxation-Time (MRT) model: poboljšava stabilnost i točnost u odnosu na Single-Relaxation-Time (SRT) model.

### 4.3.1. OpenLB implementacija

OpenLB Krause et al. (2020) je paket otvorenog koda koji implementira LBM za simulaciju dinamike fluida. Pruža fleksibilne alate za prilagodbu specifičnim simulacijskim potrebama L. Hegele (2018). Na slici 4.3 prikazan je diskretizirani lattice model koji će kasnije biti dodatno opisan i korišten u daljnim razmatranjima i simulacijama.



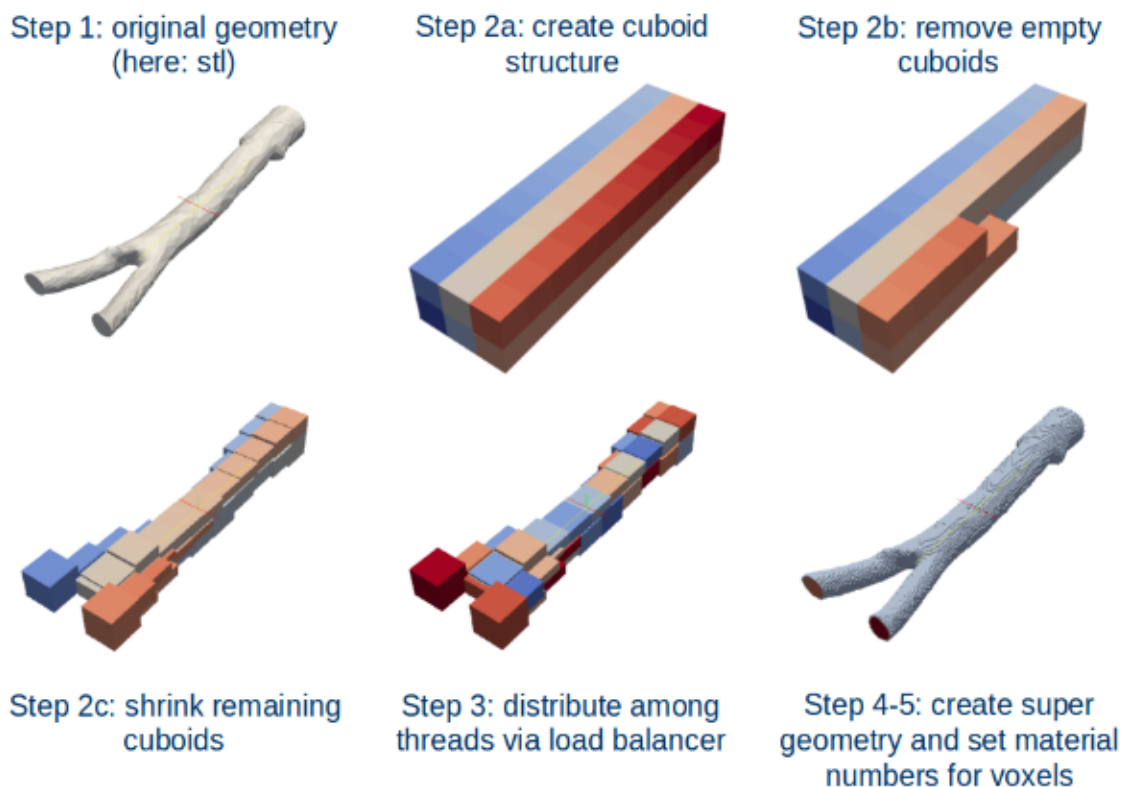
**Slika 4.3:** Shema rešetke u LBM simulacijama.

#### Dobre strane LBM-a

LBM ima nekoliko ključnih prednosti u usporedbi s tradicionalnim metodama poput FVM-a i FEM-a. Jedna od najvažnijih je njegova prirodna sposobnost za paralelizaciju. Budući da su izračuni unutar svakog čvora rešetke lokalni, LBM je vrlo efikasan za implementaciju na modernim paralelnim računalnim arhitekturama, uključujući GPU-ove (S. Kollmannsberger, 2017).

Druga prednost LBM-a je jednostavnost u rukovanju složenim rubnim uvjetima i geometrijama. Za razliku od FVM-a i FEM-a, LBM se temelji na jednostavnim pravokutnim rešetkama, što olakšava implementaciju i smanjuje kompleksnost generiranja mreže. Ovaj

pristup također omogućava jednostavno rukovanje s pokretnim granicama i višefaznim tokovima (S. Chen, 1998). Izrada 3D lattice-a iz kompleksne geometrije je prikazan prema dokumentaciji (str. 35) openLB-a na slici 4.4.



**Slika 4.4:** Izrada lattice-a iz kompleksne geometrije .stl datoteke

OpenLB je softverski paket (biblioteka) otvorenog koda koji implementira LBM. Pruža korisnicima alate za učinkovitu i fleksibilnu simulaciju dinamike fluida koristeći LBM pristup. OpenLB podržava različite fizikalne modele i omogućava jednostavnu prilagodbu i proširenje L. Hegele (2018).

### Loše strane LBM-a

Iako LBM ima mnoge prednosti, postoje i određeni nedostaci. Jedan od glavnih problema je numerička stabilnost. LBM može biti osjetljiv na različite parametre simulacije, kao što su veličina rešetke i vremenski korak. Nepravilno odabrani parametri mogu dovesti do nestabilnosti i netočnosti u simulacijama (C. K. Aidun, 2010).

Drugi nedostatak je relativno visoka računaska složenost za simulacije visoko turbulentnih tokova. Iako LBM može učinkovito simulirati laminarne tokove, zahtjevi za računalnim resursima znatno rastu s povećanjem Reynoldsovog broja, što može ograničiti primjenu LBM-a u simulacijama visoko turbulentnih tokova (Kruger i H. Kusumaatmaja, 2017).

## 4.4. Usporedba LBM-a, FVM-a i FEM-a

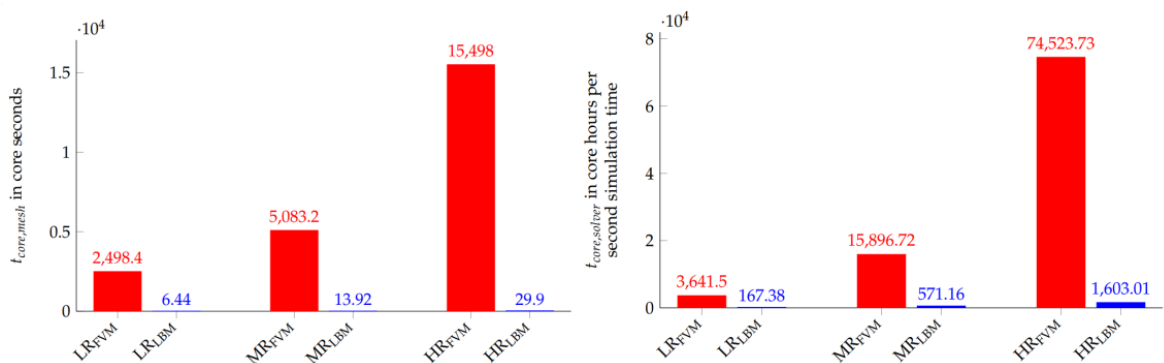
Kada se uspoređuju LBM s FVM-om i FEM-om, može se identificirati nekoliko ključnih točaka.

LBM se izdvaja svojom jednostavnošću implementacije i efikasnošću na paralelnim računalnim arhitekturama. Ovo ga čini atraktivnim za simulacije koje zahtijevaju brzu obradu velikih količina podataka, kao što su simulacije u realnom vremenu ili optimizacije dizajna (Succi, 2008).

FVM je izuzetno koristan zbog svoje konzervativne prirode i fleksibilnosti u radu s nestrukturiranim mrežama. Međutim, može imati problema s točnošću u složenim geometrijama i zahtjeva pažljivo postavljanje mreže za osiguranje stabilnosti (H. K. Versteeg, 2007).

FEM nudi visoku točnost i fleksibilnost u modeliranju složenih problema, ali može biti računski intenzivan i zahtjeva veliko iskustvo za pravilno postavljanje mreže i rubnih uvjeta. Ova metoda je posebno korisna za probleme s kompleksnim materijalnim svojstvima i geometrijama (O. C. Zienkiewicz, 2005).

Slijedom ovih razmatranja za daljnji je rad odabrana LBM metoda implementirana u openLB biblioteci. Prednosti nad drugim metodama izložili su Mathias J. Krause, Adrian Kummerländer i Stephan Simonis, a zaključci njihova predavanja više su no očiti. Izvor je podataka rad Marc Haussmann et al. (2020) gdje su detaljno opisane prednosti u izvođenju LBM-a koristeći openLB nad FVM-om korištenjem openFOAM-a. Jedna od usporedbi prikazna je na slici 4.5. Te je navedeno kako je izrada mreže za rješavanje u OpenLB-u oko 424 puta brža no u openFOAM-u dok je simulacija oko 32 puta brža uz zadržavanje podjednake točnosti rezultata.



Slika 4.5: Usporedba brzine izvođenja LBM-a i FEM-a z aisti testni slučaj

LBM je vrlo učinkovit za simulacije s jednostavnim geometrijama i gdje je potrebna brza paralelizacija. Tipični primjeri uključuju mikrofluidi, biomehanika i tokove u poroznim medijima. OpenLB omogućava jednostavnu implementaciju ovih simulacija i pruža alate za prilagodbu specifičnim zahtjevima (L. Hegele, 2018). Rad C. K. Aidun (2010) detaljno uspoređuje LBM simulacije s eksperimentalnim podacima za tokove u složenim geometrijama.

LBM se pokazao vrlo prikladnim za simulacije mikrofluidike i kompleksnih tokova uz dobru korelaciju s eksperimentalnim podacima.

FVM se često koristi u industrijskim aplikacijama gdje je konzervacija masa, količine gibanja i energije kritična. OpenFOAM pruža širok spektar mogućnosti za prilagodbu različitim industrijskim potrebama, uključujući aerodinamiku, prijenos topline i više-fazne tokove (Jasak i A. Jemcov, 2007). Rad H. K. Versteeg (2007) pruža usporedbu FVM simulacija s eksperimentalnim podacima za turbulentne tokove. Pokazano je da FVM može precizno predvidjeti ključne karakteristike toka uz pravilnu kalibraciju modela turbulencije.

FEM se široko koristi u građevinarstvu, mehanici čvrstih tijela i problemima s kompleksnim materijalnim svojstvima. Elmer je posebno pogodan za termomehaničke simulacije i modeliranje složenih interakcija u sustavima (M. Malinen, 2008). Rad O. C. Zienkiewicz (2005) uspoređuje FEM simulacije s eksperimentalnim rezultatima u strujanjima oko kompleksnih geometrija. FEM se pokazao vrlo učinkovitim za detaljno predviđanje ponašanja fluida uz visoku razinu točnosti.

# 5. Lattice-Boltzmanova metoda

## 5.1. Razvoj Lattice-Boltzmann metode

Lattice-Boltzmann metoda (LBM) potječe iz ranih radova na staničnim automatima za simulaciju fluida, osobito radova Frisch i B. Hasslacher (1986), koji su uveli Lattice Gas Cellular Automaton (LGCA). LGCA je bila prva metoda koja je koristila diskretni prostor i vrijeme za simulaciju fluidne dinamike, ali je patila od velikih statističkih fluktuacija i problema s nestabilnošću.

LBM se pojavila krajem 1980-ih i početkom 1990-ih kao poboljšanje LGCA, koje je zamijenilo binarna stanja čestica u distribucijskim funkcijama. Prvi radovi koji su opisali osnovne principe LBM su radovi F. J. Higuera (1988) te G. R. McNamara (1988). LBM je brzo stekla popularnost zbog svoje jednostavnosti implementacije, paralelizacije i sposobnosti da prirodno rješava složene granične uvjete.

## 5.2. Osnovne definicije LBM-a

Daljnji izvod LBM-a bit će baziran na Wagner (2008).

### 5.2.1. Distribucijska funkcija i osnovne jednadžbe

U LBM-u, fluid je opisan distribucijskom funkcijom  $f_i(\mathbf{x}, t)$  koja predstavlja gustoću čestica s brzinom  $\mathbf{e}_i$  u položaju  $\mathbf{x}$  i vremenu  $t$ . Evolucija distribucijske funkcije je dana Lattice Boltzmann jednadžbom:

$$f_i(\mathbf{x} + \mathbf{e}_i \Delta t, t + \Delta t) - f_i(\mathbf{x}, t) = -\frac{\Delta t}{\tau} [f_i(\mathbf{x}, t) - f_i^{\text{eq}}(\mathbf{x}, t)], \quad (5.1)$$

gdje je  $\tau$  vrijeme relaksacije, a  $f_i^{\text{eq}}$  ekvilibrij distribucijska funkcija.

Da bismo razumjeli ekvilibrij distribucijsku funkciju, možemo koristiti pristup iz statističke mehanike. Ekvilibrij distribucijska funkcija  $f^{\text{eq}}$  minimizira Boltzmannov H-funktional:

$$H(t) = \int dx dv f(x, v, t) \log f(x, v, t). \quad (5.2)$$

Ovo minimiziranje se provodi uz uvođenje Lagrangeovih multiplikatora kako bi se osigurale konzervacije mase, momenta i energije:

$$\begin{aligned}
H(t) = & \int dx dv f(x, v, t) \log f(x, v, t) \\
& - \lambda_1 \left( nV - \int dx dv f(x, v, t) \right) \\
& - \lambda_2 \alpha \left( nu_\alpha V - \int dx dv f(x, v, t) v_\alpha \right) \\
& - \lambda_3 \left( n\epsilon V - \int dx dv f(x, v, t) \frac{v^2}{2} \right).
\end{aligned} \tag{5.3}$$

Minimizacija H-funkcionala daje:

$$0 = \left. \frac{\delta H(t)}{\delta f} \right|_{f=f_0} = 1 + \log(f_0) + \lambda_1 + \lambda_2 \alpha v_\alpha + \lambda_3 \frac{v^2}{2}. \tag{5.4}$$

Iz toga slijedi da je ekvilibrir distribucijska funkcija  $f_0$ :

$$f_0 = \exp \left[ -1 - \lambda_1 + \lambda_2 v_\alpha + \lambda_3 \frac{v^2}{2} \right]. \tag{5.5}$$

Uz uvođenje novih Lagrangeovih multiplikatora  $a$ ,  $b_\alpha$  i  $c$ , ovo se može zapisati kao:

$$f_0 = a \exp \left[ \frac{(b - v)^2}{c} \right]. \tag{5.6}$$

Ova distribucijska funkcija ne ovisi o prostornim varijablama, što znači da je rješenje homogeno. Rješenje za Lagrangeove multiplikatore  $a$ ,  $b_\alpha$  i  $c$  se može naći u uvjeta očuvanja:

$$nV = \int dx dv f_0, \tag{5.7}$$

$$nu_\alpha V = \int dx dv f_0 v_\alpha, \tag{5.8}$$

$$n\epsilon V = \int dx dv f_0 \frac{v^2}{2}. \tag{5.9}$$

Rješavanjem ovih jednačbi dobivamo Maxwell-Boltzmannovu distribuciju:

$$f_0 = n \left( \frac{1}{2\pi\theta} \right)^{3/2} \exp \left[ -\frac{(v - u)^2}{2\theta} \right], \tag{5.10}$$

gdje su  $n$  gustoća,  $u$  brzina, i  $\theta$  temperatura fluida.

### 5.3. Zakoni očuvanja i derivacija transportnih jednadžbi

Za potpuno razumijevanje transportnih jednadžbi izvedenih iz Boltzmannove metode potrebno je pojedinačno razumjeti i osigurati očuvanje momenta kao i očuvanje mase te energije. Bez ovih jednadžbi nije moguće osigurati sigurnost u rješenja koja proizlaze iz danih rezultata te će svaki od zakona očuvanja biti pobliže opisan.

#### Očuvanje mase

Integriranjem Boltzmannove jednadžbe preko svih brzina dobiva se jednadžba očuvanja mase:

$$\partial_t \int dv f + \partial_\alpha \int dv f v_\alpha + F \int dv \partial_v f = \frac{1}{\tau} \int dv (f^0 - f) \quad (5.11)$$

S obzirom na to da masa mora biti očuvana, članovi na desnoj strani jednadžbe moraju težiti ka nuli te za tim slijedi:

$$\partial_t n + \partial_\alpha (n u_\alpha) = 0 \quad (5.12)$$

Ova jednadžba je poznata kao jednadžba kontinuiteta i već je opisana 4.1 te osigurava očuvanje mase.

#### Očuvanje momenta

Pomnoži li se Boltzmannove jednadžbe s brzinom  $v_\alpha$  i integriranjem preko svih brzina dobivena je jednadžba očuvanja momenta:

$$\partial_t \int dv f v_\alpha + \partial_\beta \int dv f v_\alpha v_\beta + F_\beta \int dv \partial_{v_\beta} f v_\alpha = \frac{1}{\tau} \int dv (f^0 - f) v_\alpha \quad (5.13)$$

Nakon pojednostavljenja koristeći aproksimaciju  $f = f^0 - \tau(\partial_t f + v \partial_x f + F \partial_v f)$ :

$$\partial_t (n u_\alpha) + \partial_\beta (n u_\alpha u_\beta) = -\partial_\alpha (n \theta) + n F_\alpha \quad (5.14)$$

Ova jednadžba je poznata kao Eulerova jednadžba koja opisuje očuvanje momenta. Koristeći jednadžbu kontinuiteta, moguće je zapisati u obliku:

$$\partial_t u_\alpha + u_\beta \partial_\beta u_\alpha = -\frac{1}{n} \partial_\alpha (n \theta) + F_\alpha \quad (5.15)$$

Za izvod Navier-Stokesove jednadžbe, potrebno je evaluirati više redove derivacija:

$$n \partial_t u_\alpha + n u_\beta \partial_\beta u_\alpha = -\partial_\alpha (n \theta) + n F_\alpha + \partial_\beta [\eta (\partial_\beta u_\alpha + \partial_\alpha u_\beta - \frac{2}{3} \partial_\gamma u_\gamma \delta_{\alpha\beta})] \quad (5.16)$$

gdje je  $\eta = n \theta \tau$  viskozitet.



## Očuvanje energije

Množenjem Boltzmannove jednadžbe s  $(v - u)^2$  i integriranjem preko svih brzina slijedi jednadžba očuvanja energije:

$$\partial_t \int dv f (v - u)^2 + \partial_\alpha \int dv f v_\alpha (v - u)^2 + 2n\theta \partial_\alpha u_\alpha - \tau \partial_\alpha u_\beta (\partial_\alpha u_\beta + \partial_\beta u_\alpha - \frac{2}{3} \partial_\gamma u_\gamma \delta_{\alpha\beta}) = 0 \quad (5.17)$$

Koristeći aproksimaciju  $f = f^0 - \tau(\partial_t f + v \partial_x f + F \partial_v f)$  slijedi:

$$\partial_t \theta + u_\alpha \partial_\alpha \theta = -\frac{2}{3} \partial_\alpha u_\alpha \theta + \frac{1}{n} \partial_\alpha \left( \frac{5n\theta}{3} \partial_\alpha \theta \right) + \tau \partial_\alpha u_\beta \left( \partial_\alpha u_\beta + \partial_\beta u_\alpha - \frac{2}{3} \partial_\gamma u_\gamma \delta_{\alpha\beta} \right) \quad (5.18)$$

Ova jednadžba je poznata kao jednadžba prijenosa topline. Gore navedeni i izvedeni zakoni očuvanja su ključni za derivaciju hidrodinamičkih jednadžbi iz Boltzmannove jednadžbe, pružajući temelj za simulaciju fluidnih tokova pomoću metode Lattice Boltzmann.

## 5.4. Dimenzionalna Analiza

Dimenzionalna analiza je ključni alat u razumijevanju i modeliranju fizičkih fenomena, uključujući fluidnu dinamiku. Korištenjem dimenzionalne analize, možemo smanjiti broj promjenjivih u problemu i identificirati bezdimenzionalne brojeve koji karakteriziraju sustav.

### 5.4.1. Karakteristične Duljine, Vremena i Brzine

U kontekstu fluidne dinamike, osnovne fizičke veličine koje koristimo za skaliranje su:

- Karakteristična duljina ( $L$ ): Ovo je tipična duljina skale problema, npr. promjer cijevi, duljina krila aviona ili veličina rešetke u Lattice-Boltzmann metodi.
- Karakteristično vrijeme ( $T$ ): Ovo je tipično vrijeme skale, koje može biti određeno brzinom karakterističnih fenomena u sustavu, npr. vrijeme prolaska čestice kroz određeni dio sustava.
- Karakteristična brzina ( $U$ ): Tipična brzina skale, npr. srednja brzina fluida u sustavu.

### 5.4.2. Definiranje Bezdimezionalnih veličina

Korištenjem gore navedenih karakterističnih veličina, fizičke veličine sustava mogu se izraziti u bezdimenzionalnom obliku. Na primjer:

- Bezdimenzionalna duljina ( $x^*$ ):

$$x^* = \frac{x}{L} \quad (5.19)$$

- Bezdimenzionalno vrijeme ( $t^*$ ):

$$t^* = \frac{t}{T} \quad (5.20)$$

- Bezdimenzionalna brzina ( $u^*$ ):

$$u^* = \frac{u}{U} \quad (5.21)$$

### 5.4.3. Bezdimenzionalni Brojevi

Dimenzionalna analiza omogućava identificiranje ključnih bezdimenzionalnih brojeva koji karakteriziraju tok fluida. Neki od najvažnijih brojeva uključuju:

- Reynoldsov broj (Re)::

$$\text{Re} = \frac{UL}{\nu} \quad (5.22)$$

- Strohalov broj (St):

$$\text{St} = \frac{L}{UT} \quad (5.23)$$

- Bezdimenzionalna brzina ( $u^*$ ):

$$u^* = \frac{u}{U} \quad (5.24)$$

gdje je  $\nu$  kinematička viskoznost fluida. Reynoldsov broj opisuje odnos između inercijskih i viskoznih sila u fluidu. Strohalov broj opisuje odnos između karakteristične duljine i brzine te vremena.

### 5.4.4. Primjena u Lattice-Boltzmann metodi (LBM)

U Lattice Boltzmann metodi (LBM), dimenzionalna analiza igra ključnu ulogu u osiguravanju točnosti i stabilnosti simulacija. Ključne bezdimenzionalne veličine u LBM uključuju:

- Bezdimenzionalna brzina mreže ( $u^*$ ):

$$u^* = \frac{u}{c_s} \quad (5.25)$$

gdje je  $c_s$  brzina zvuka u diskretnoj rešetki

- Bezdimenzionalno vrijeme relaksacije ( $\tau^*$ ):

$$\tau^* = \frac{\tau}{\Delta t} \quad (5.26)$$

gdje je  $\tau$  vrijeme relaksacije u LBM modelu, a  $\Delta t$  vremenski korak

U implementaciji OpenLB biblioteke, objekti kao što je `converter` koriste se za skaliranje fizičkih veličina na rešetku LBM-a. Na primjer, karakteristične duljine i brzine fizičkog sustava prevode se u diskretne jedinice korištenjem odgovarajućih faktora skaliranja: `UnitConverterFromResolutionAndRelaxationTime converter`

```
N // Resolution: number of lattice units per characteristic length
L // Physical characteristic length
U // Physical characteristic velocity
Re // Reynolds number
tau // Relaxation time
```

Ovdje ‘N’ označava rezoluciju ili broj rešetkastih jedinica po karakterističnoj duljini. Ovaj pristup osigurava da su svi bezdimenzionalni brojevi pravilno skalirani i da se fizičke jedinice točno preslikavaju na diskretne jedinice korištene u LBM simulacijama.

### 5.4.5. Implementacija skaliranja u OpenLB-u

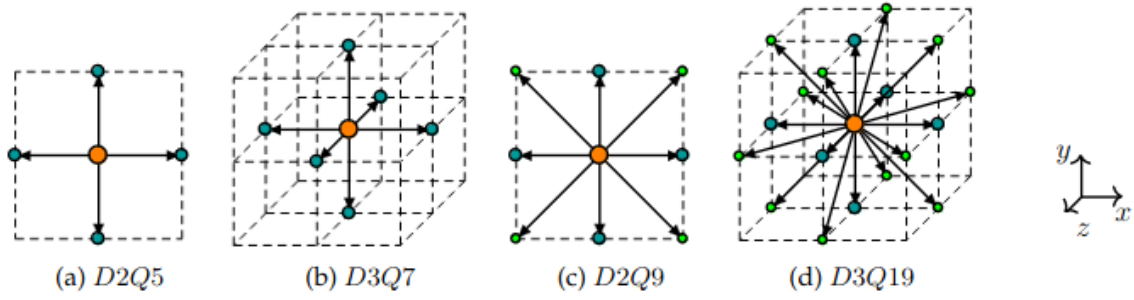
OpenLB koristi objekt `converter` za rukovanje skaliranjem simulacija. Ovaj objekt omogućuje korisnicima da lako pretvaraju fizičke veličine u odgovarajuće dimenzionalne jedinice korištene u LBM simulacijama.

```
UnitConverterFromResolutionAndRelaxationTime<T, DESCRIPTOR> const converter(
int {N}, // resolution: number of voxels per charPhysL
(T) 0.8, // latticeRelaxationTime: relaxation time, have to be greater than 0.5!
(T) 2.0*radiusCylinder, // charPhysLength: reference length of simulation geometry
(T) 1.5, // charPhysVelocity: maximal/highest expected velocity during simulation in __ m / s__
(T) 0.2*2.*radiusCylinder/Re, // physViscosity: physical kinematic viscosity in __ m^2 / s__
(T) 1.239 // physDensity: physical density in __ kg / m^3__
);
```

**Slika 5.1:** Dijagram objekta `converter` u OpenLB-u koji prikazuje proces skaliranja fizičkih veličina za jedan od slučajeva simulacije.

## 5.5. Metode Diskretizacije

Diskretizacija u LBM-u obuhvaća diskretizaciju prostora, vremena i brzine. Diskretizacija prostora uključuje podjelu domene na rešetku, dok diskretizacija vremena uključuje definiranje diskretnih vremenskih koraka. Diskretizacija brzine uključuje odabir diskretnih brzina  $e_i$  koje definiraju moguće smjerove gibanja čestica. Metode diskretizacije implementirane su kao deskriptori raspoloživih za korištenje u jednoj, dvije ili tri dimenzije ( $D$ ) te određene brojem susjednih ćelija na koje se distribucija primjenjuje ( $Q$ ).



**Slika 5.2:** Shema deskriptora rešetki i diskretnih brzina, (a, b):  $DdQ(2d + 1)$  za  $d = 2, 3$ ; (c):  $D2Q9$ ; (d):  $D3Q19$ . Bojenje se odnosi na odgovarajuću energetsku ljusku, gdje narančasta, cijan i zelena označavaju prvo nulu prema Simonis (2023)

### 5.5.1. D2Q9 Metoda

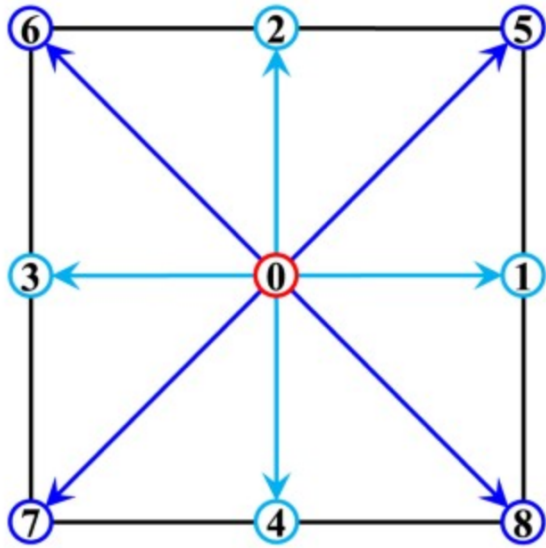
$D2Q9$  metoda je najčešće korištena diskretizacija brzine u dvodimenzionalnim LBM simulacijama. Ova metoda koristi devet diskretnih brzina:

$$\mathbf{e}_i = \begin{cases} (0, 0) & i = 0, \\ (1, 0), (-1, 0), (0, 1), (0, -1) & i = 1, 2, 3, 4, \\ (1, 1), (-1, 1), (-1, -1), (1, -1) & i = 5, 6, 7, 8, \end{cases} \quad (5.27)$$

gdje su brzine  $\mathbf{e}_i$  vektori koji definiraju smjerove gibanja čestica. Težinski faktori za  $D2Q9$  metodu su:

$$w_i = \begin{cases} 4/9 & i = 0, \\ 1/9 & i = 1, 2, 3, 4, \\ 1/36 & i = 5, 6, 7, 8. \end{cases} \quad (5.28)$$

$D2Q9$  metoda omogućuje efikasnu simulaciju dvodimenzionalnih tokova s visokim stupnjem točnosti, a model interakcije među ćelijama prema prethodnim oznakama susjednih ćelija prikazana je na slici 5.3. OpenLB implementira  $D2Q9$  metodu pružajući fleksibilnost u definiranju graničnih uvjeta i fizikalnih modela. OpenLB je optimiziran za paralelne simulacije, omogućujući brzu obradu velikih problema (L. Hegele, 2018). OpenLB implementira  $D2Q9$  metodu pružajući fleksibilnost u definiranju graničnih uvjeta i fizikalnih modela.



Slika 5.3: Model diskretizacije i oznake susjednih ćelije

## **6. Trenutni načini simulacije**

### **6.1. Numerička studija elektroaerodinamičke sile i struje u sustavima s emiterom i kolektorom**

Studija koju su proveli Christian Coseru i Plourabou (2021) istražuje složenu dinamiku elektroaerodinamičkih (EAD) sila i rezultirajućeg ionskog vjetra u sustavima s emiterom i kolektorom. Rezultati ove studije će biti korišteni za evaluaciju rješenja dobivenog LBM-om. Primarni fokus studije bio je razumjeti ponašanje ionskog vjetra generiranog EAD silama. Autori su koristili numerički pristup za simulaciju interakcija unutar sustava emiter/kolektor. Fokus će biti stavljen na postav s jednim emiterom i jednim kolektorom zbog računalne jednostavnosti.

#### **6.1.1. Elektroaerodinamičke sile**

Studija pokazuje da se EAD sile generiraju zbog kretanja iona u električnom polju. Električno polje se stvara između emitera, koji otpušta ione, i kolektora, koji ih privlači. Ovaj proces generira potisnu silu koja je ključna za primjene u sustavima za propulziju. Distribucija električnog polja igra značajnu ulogu u generiranju ove sile.

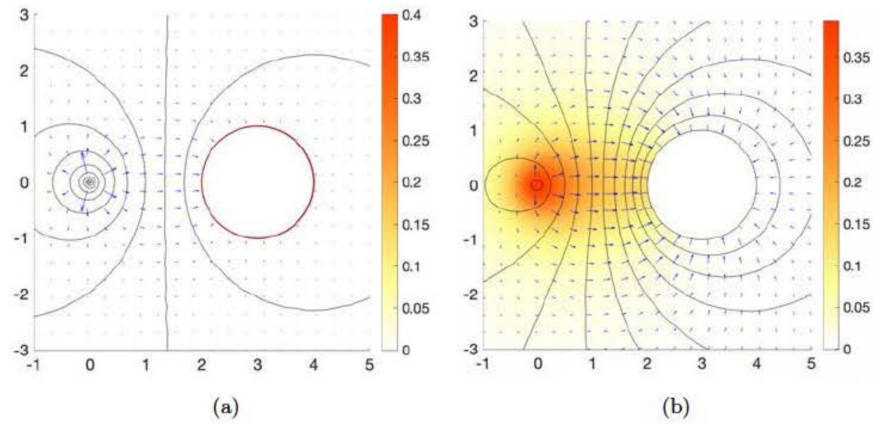


Figure 3: Ion density  $\hat{I}$  (color levels), electric potential  $\hat{\phi}$  (isolines) and electric field  $\hat{\mathbf{E}}$  (vectors) corresponding to 1E/1C case with  $r_e = 50\mu m$ ,  $R_{col} = 15mm$ ,  $D = 30mm$ , for (a)  $V = \varphi_s = 6.47kV$  very close, and above the onset voltage (b)  $V = 20kV$  where the corona discharge is triggered.

**Slika 6.1:** Distribucija električnog polja između emitera i kolektora prema radu Christian Coseru i Plourabou (2021).

### 6.1.2. Ionski vjetar i parametri sustava

Ionski vjetar nastaje kao rezultat kretanja iona pod utjecajem električnog polja. Ova struja iona stvara aerodinamički potisak koji može biti korišten u različitim tehnološkim primjenama, poput tihih i efikasnih pogonskih sustava. Autori su kvantificirali brzinu i smjer ionskog vjetra u različitim konfiguracijama sustava. Studija također istražuje utjecaj različitih parametara na EAD sile i ionski vjetar, uključujući razmak između emitera i kolektora, napon primijenjen na sustav, te geometrijske karakteristike emitera i kolektora. Utvrđeno je da povećanje napona dovodi do jačih EAD sila i većeg ionskog vjetra, dok razmak između emitera i kolektora značajno utječe na efikasnost sustava. U nastavku rada bit će uspoređeni samo pojedini slučajevi.

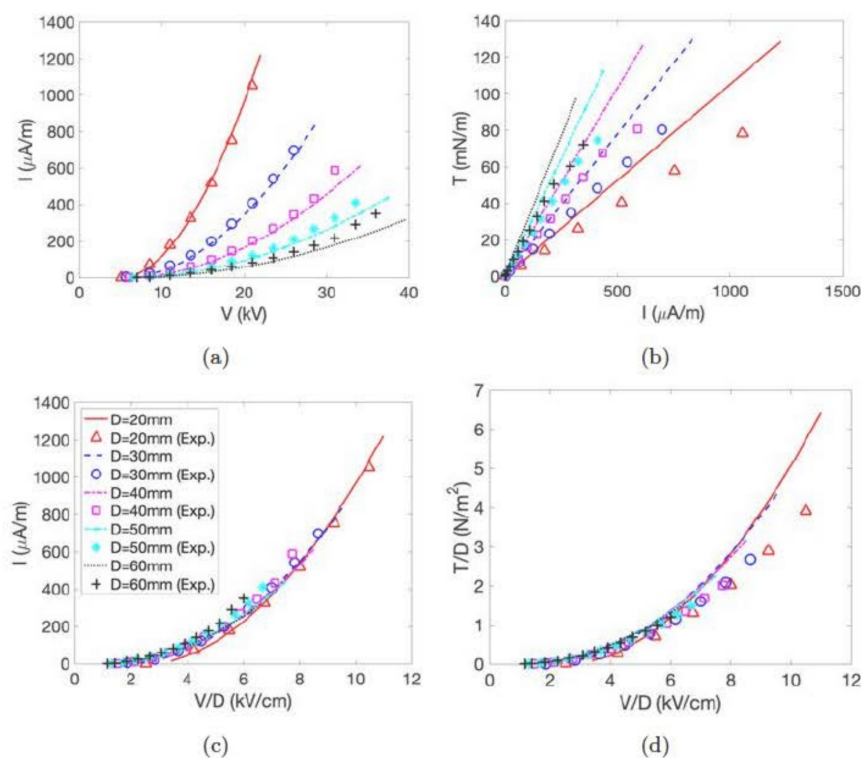


Figure 11: Results for 1E1C configurations with  $r_c = 5\text{mm}$ ,  $r_0 = 25\mu\text{m}$  and several values of gap  $D$ . Comparison with experimental data by Monrolin et al. All panels a,b,c,d share the same legend as (c), hence, not duplicated.

**Slika 6.2:** Usporedba potiska, napona i struje sa starnim podacima mjerjenima Nicolas Monrolin i Praud (2017).

## 6.2. Uloga nelinearnih sila u elektrostatičkom propeleru pokretanom atmosferskim DC koronalnim pražnjenjima

Rad Sohbatzadeh i H. Soltani (2021) istražuje ulogu nelinearnih sila na tijelo u elektrostatičkim propelerima koji koriste atmosferski DC korona pražnjenja za generiranje potiska. Pregled ključnih nalaza studije i razmatranja bit će diskutirana u nastavku kako se ovi nalazi mogu primijeniti u razvoju sličnih sustava uz pomoć LBM-a u OpenLB-u.

### 6.2.1. Elektrodinamičke sile

Rad se usredotočuje na analizu nelinearnih sila koje se pojavljuju u elektrostatičkim propulzorima kako pozitivnog koronalnog izboja tako i negativnog. U ovome će rad daljnje simulacije biti usredotočen na pozitivni izboj uslijed veće efikasnosti, a time i primjenjivosti istoga kako je zaključeno prema Nicolas Monrolin i Praud (2017). Studija pokazuje da nelinearne sile igraju ulogu u učinkovitosti elektrostatičkih propulzora, no njihov utjecaj je relativno malen gledano na ukupnoj domeni potrebnoj za razumijevanje tokova zraka.



Nelinearne sile nastaju zbog složenih interakcija između električnog polja i iona stvorenih korona pražnjenjem. Ove sile mogu značajno utjecati na generirani potisak i učinkovitost propulzije. Efekti nelinearnih sila slijede iz jakosti električnog polja na površinama malog promjera te time omogućuju veću produkciju iona u fluidu. Daljnjim interakcijama tako nastalih iona na ukupnoj domeni simulacije moguće je aproksimirati njihov utjecaj uslijed provedenih sudara.

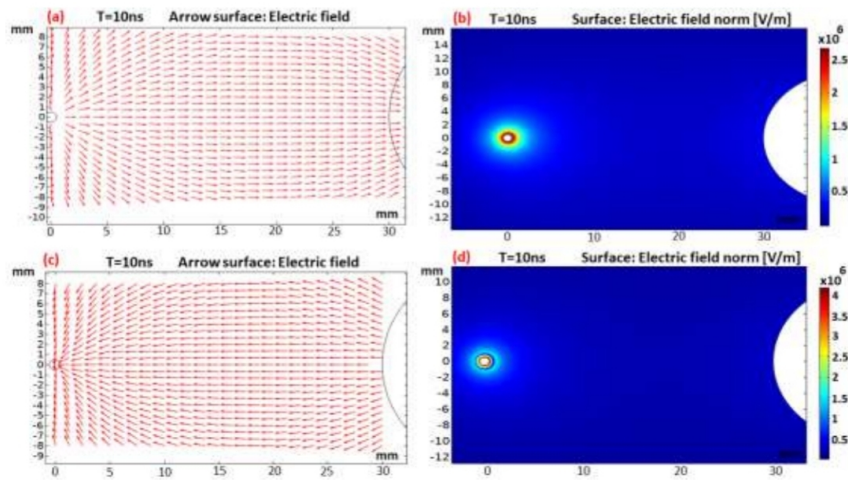
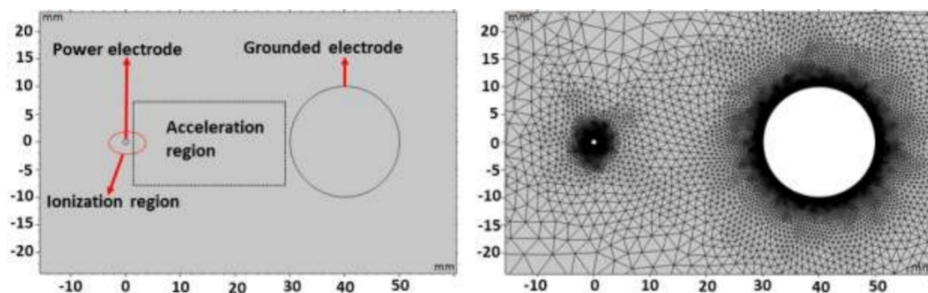


Figure 8. 2-D Plots of the electric field distributions (a) vector field (b) electric field magnitude in the positive corona discharge. (c) vector field, and (d) electric field magnitude for negative corona.

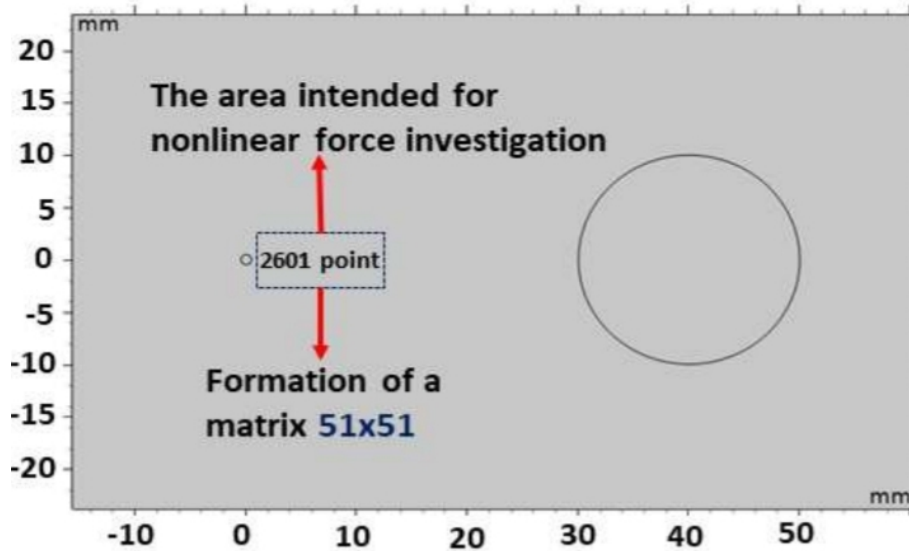
Slika 6.3: Primjer električnog polja između elektroda.

## 6.2.2. Ionski vjetar i parametri sustava

Utvrđeno je da nelinearne sile utječu na dizajn elektrostatičkih propulzora samo u domeni izvedbe elektroda. Optimalni dizajn mora uzeti u obzir ove sile kako bi se postigla maksimalna učinkovitost tj. potrebno je poznavanje regije koja je zadužena za generiranje plazme. Studija predlaže promjene u geometriji i električnom polju kako bi se poboljšale performanse propulzora.



Slika 6.4: Geometrija i domena za provođenje simulacije.



Slika 6.5: Geometrija i domena utjecaja nelinearne sile.

### 6.2.3. Numerička simulacija i eksperimentalni rezultati

Rad koristi numeričke simulacije i eksperimentalne rezultate kako bi prikazao utjecaj nelinearnih sila na performanse propulzora. Simulacije pokazuju kako različite konfiguracije korona pražnjenja mogu utjecati na raspodjelu nelinearnih sila i učinkovitost propulzora. Na slikama 6.6 i 6.7 prikazani su utjecaji linearne i nelinearne sile.

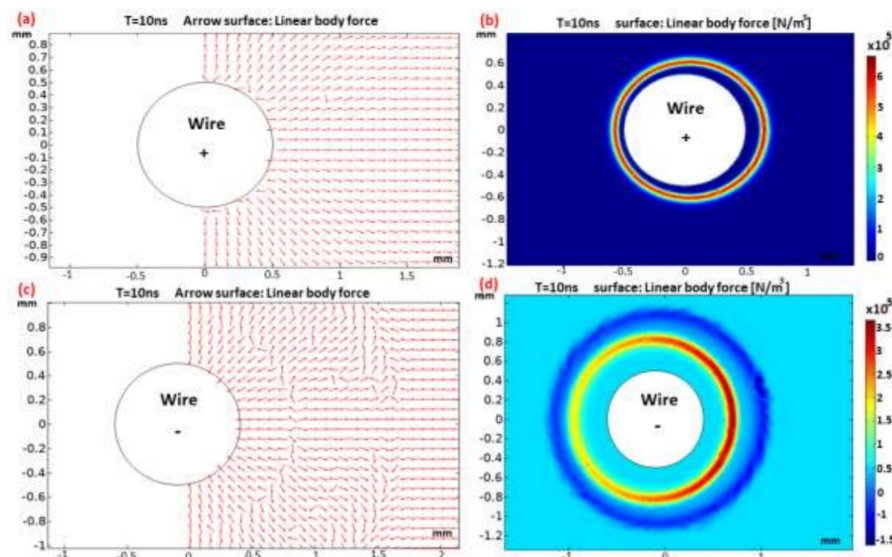


Figure 11. 2-D Plots of the linear body force term (a) vector field, (b) magnitude of the linear term in the positive corona discharge. (c) vector field, and (d) magnitude of the linear body force for negative corona.

Slika 6.6: Rezultati numeričke simulacije linearnih sila u elektrostatičkom propulzoru oko elektrode.

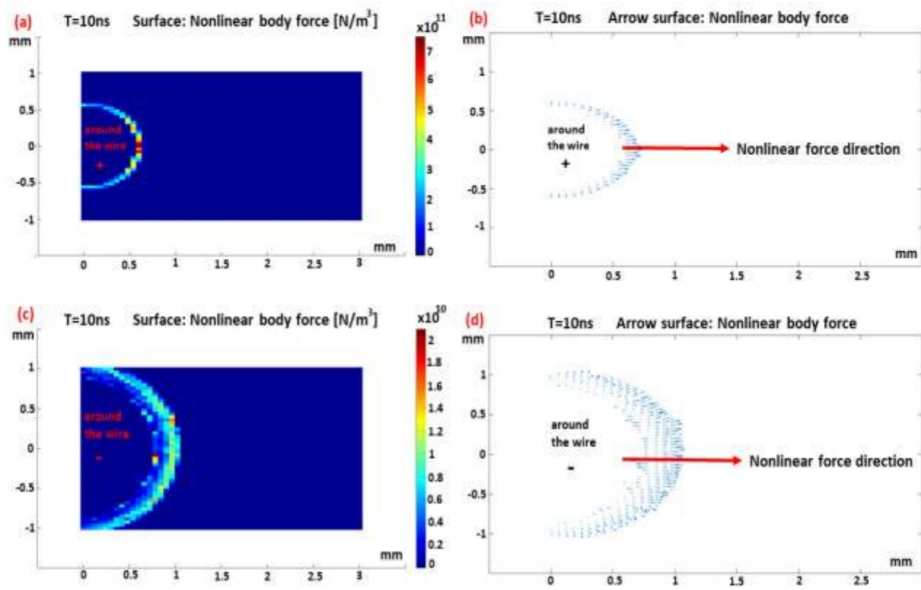


Figure 12. 2-D plots for the magnitude (a) and direction of the nonlinear body force (b) in the positive corona discharge. (c) magnitude, and (d) direction of the nonlinear force in the negative corona discharge.

**Slika 6.7:** Rezultati numeričke simulacije nelinearnih sila u elektrostatičkom propulzoru oko elektrode.

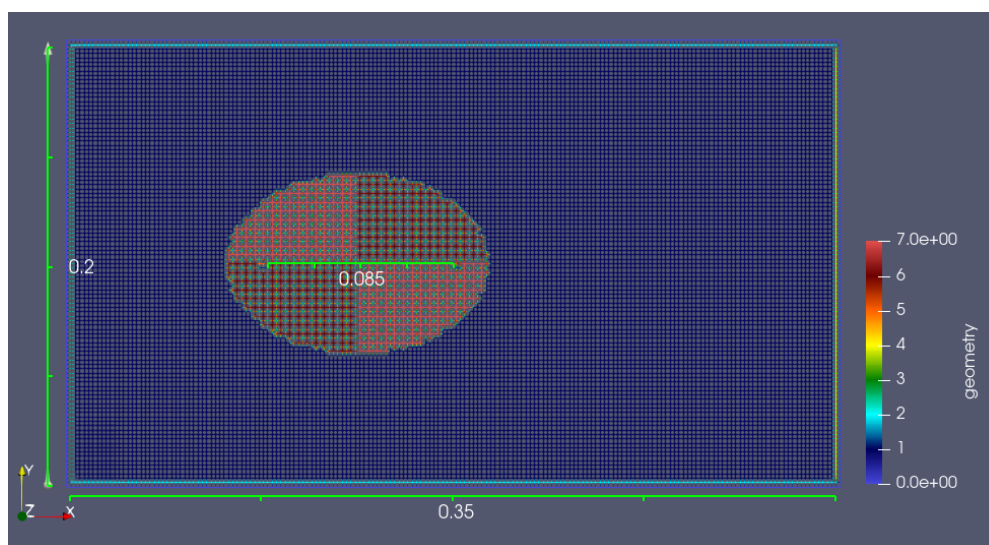
# 7. Simulacija

Perma navedenim metodama i radovima bit će izvedena simulacija toka zraka oko dviju elektroda pozitivnog koronalnog izboja. Sila će biti aproksimirana analitički kao potisak na elektrodu te uspoređena s očekivanim vrijednostima kako bi bilo moguće odrediti kvalitetu simulacije kao i njenu efikasnost.

## 7.1. Postavljanje domene

Kako bi bilo moguće provesti numeričku simulaciju nužno je odrediti područje računanja i evaluacije vrijednosti jednadžbi koje opisuju fizikalne efekte. Dimenzija domene je postavljan na 35x20 cm. Debljina vodiča je postavljena na 0.2 mm. Materijal koji ispunjava domenu je zrak, a definiran je pri standardnim atmosferskim uvjetima tako da mu je:

- gustoća:  $1.239 \text{ kg/m}^3$
- permeabilnost ( $\eta_r$ ): 1.000 589 86
- vlažnost: 55%
- mobilnost elektrona ( $\mu$ ):  $1.8 \frac{10^{-4} \text{ m}^2}{\text{Vs}}$



Slika 7.1: Simulacijska domena s rubnim uvjetima i dimenzijama

- materijal 0: unutar elektroda -nema proračuna
- materijal 1: fluid bez utjecaja vanjske sile
- materijal 2: rubovi domene (no-slip/bounce zidovi)
- materijal 3: upuh
- materijal 4: ispuh
- materijal 5: rubovi elektroda (bounce zidovi)
- materijal 6: fluid sa silom (pozitivan x i pozitivan y smjer)
- materijal 7: fluid sa silom (pozitivan x i negativan y smjer)

## 7.2. Električna svojstva unutar domene

Domena je postavljena tako da je očekivano strujanje u smjeru pozitivne x osi. Kako bi bilo moguće izvesti pozitivan koronalni izboj nužno je postaviti pozitivan napon na prvu elektrodu (manje vrijednosti u x osi). Taj je napon postavljen je na 12 kV pošto je to vrijednost koja je veoma blizu naponu stvaranja električnog luka u izboju koji prestaje biti koronalni u danim uvjetima fluida unutar domene. Električno polje prikazano je na slici 7.2 te je električni potencijal prikazan na slici 7.3. Smjer djelovanja električnog polja prikazan je na slici 7.4.

### 7.2.1. Izračun električnog potencijala i polja

U ovom radu implementiran je numerički model za izračun električnog potencijala i električnog polja između dviju elektroda u 2D domeni. Elektrode su postavljene na određene pozicije unutar pravokutnog područja, a razlika potencijala između njih je poznata. Korištene su osnovne fizikalne jednadžbe i filtracija pojedinih vrijednosti za izračun potencijala i polja te smo rezultate prikazali u grafičkom obliku. Domena i geometrija su strukturirane u kodu prema isječku:

```

1 # Given values
2 voltage_difference = 12e3 # 12 kV in volts
3 distance_between_electrodes = 7.5e-2 # 7.5 cm in meters
4
5 # Domain dimensions
6 domain_width = 35e-2 # 35 cm in meters
7 domain_height = 20e-2 # 20 cm in meters
8
9 # Electrode positions in cm, converted to meters
10 electrode1_x = 10e-2 # 10 cm in meters
11 electrode2_x = electrode1_x + distance_between_electrodes #
   ↪ 17.5 cm in meters
12 electrode_y = 10e-2 # 10 cm in meters
13
14 # Resolution
15 resolution = 0.5e-3 # 0.5 mm in meters
16 cell_radius = 3 # Radius of cells to exclude

```

```

17 # Create a grid of points in the 2D domain
18 x_points = np.arange(0, domain_width, resolution)
19 y_points = np.arange(0, domain_height, resolution)
20 X, Y = np.meshgrid(x_points, y_points)
21
22 # Initialize the potential array
23 V = np.zeros_like(X)
24

```

Osnovna jednadžba za električni potencijal  $V$  u blizini točkastog naboja je:

$$V = \frac{Q}{4\pi\epsilon_0 r} \quad (7.1)$$

gdje je  $Q$  količina naboja,  $\epsilon_0$  električna permitivnost vakuumu, a  $r$  udaljenost od naboja. U opisanom slučaju, elektrode smo tretirali kao izvore potencijala, a udaljenost od elektroda korištena je za računanje potencijala u svakoj točki domene.

```

1 # Calculate the electric potential for every point in the
  → domain
2 for i, y in enumerate(y_points):
3     for j, x in enumerate(x_points):
4         # Calculate distance from electrodes
5         distance_to_electrode1 = np.sqrt((x -
  → electrode1_x)**2 + (y - electrode_y)**2)
6         distance_to_electrode2 = np.sqrt((x -
  → electrode2_x)**2 + (y - electrode_y)**2)
7
8         # Calculate the potential at this point
9         if distance_to_electrode1 > 0:
10            V[i, j] += voltage_difference /
  → distance_to_electrode1
11        if distance_to_electrode2 > 0:
12            V[i, j] -= voltage_difference /
  → distance_to_electrode2

```

Filtriranje je točaka provedeno kao bi se izbjegle vrijednosti koje fizikalno nije moguće opisati, a time slijedi kako su produkt samih numeričkih nestabilnosti ove metode. Takve su vrijednosti jednostavno su isključene iz daljnjih proračuna te je usrednjena vrijednost vanjskih susjeda gledano radijalno od središta elektrode uzeta kao vrijednost u navedenoj točki.

Razlika potencijala između dviju elektroda uzrokuje raspodjelu potencijala u domeni. Korištena je sljedeća formula za izračun potencijala u svakoj točki domene:

$$V(x, y) = \frac{V_0}{r_1} - \frac{V_0}{r_2} \quad (7.2)$$

gdje su  $V_0$  razlika potencijala između elektroda,  $r_1$  udaljenost od prve elektrode, a  $r_2$  udaljenost od druge elektrode, a implementacija u kodu slijedu u nastavku:

```

1 # Mask for points within 2 cells of each electrode
2 mask = np.zeros_like(V, dtype=bool)
3 for i, y in enumerate(y_points):
4     for j, x in enumerate(x_points):
5         distance_to_electrode1 = np.sqrt((x -
  → electrode1_x)**2 + (y - electrode_y)**2)
6         distance_to_electrode2 = np.sqrt((x -
  → electrode2_x)**2 + (y - electrode_y)**2)

```



```

7         if distance_to_electrode1 <= 2 * resolution or
8           → distance_to_electrode2 <= 2 * resolution:
9             mask[i, j] = True
10        # Replace masked values with the average of the surrounding
11        → cells
12        for i, y in enumerate(y_points):
13            for j, x in enumerate(x_points):
14                if mask[i, j]:
15                    neighbors = []
16                    for di in [-1, 1]:
17                        for dj in [-1, 1]:
18                            ni, nj = i + di * cell_radius, j + dj *
19                            → cell_radius
20                            if 0 <= ni < V.shape[0] and 0 <= nj <
21                            → V.shape[1]:
22                                neighbors.append(V[ni, nj])
23                if neighbors:
24                    V[i, j] = np.mean(neighbors)

```

Električno polje  $\mathbf{E}$  povezano je s električnim potencijalom putem gradijenta:

$$\mathbf{E} = -\nabla V \quad (7.3)$$

```

1 # Calculate electric field components as negative gradient
2 → of potential
Ey, Ex = np.gradient(-V, resolution)

```

Komponente električnog polja u kartezijskom sustavu su:

$$E_x = -\frac{\partial V}{\partial x}, \quad E_y = -\frac{\partial V}{\partial y} \quad (7.4)$$

Izračun komponente električnog polja numeričkim derivacijama potencijala:

$$E_x = -\frac{V(x + \Delta x, y) - V(x, y)}{\Delta x}, \quad E_y = -\frac{V(x, y + \Delta y) - V(x, y)}{\Delta y} \quad (7.5)$$

Za prikaz veličine iznosa električnog polja korištena je logaritamsku skalu kako bi bila istaknuta razlika u intenzitetu polja te je uz to duljina vektora smjera normalizirano kako bi prikaz bio pregledniji, a bojom vektora je prezentiran iznos električnog polja u točki njegova početka:

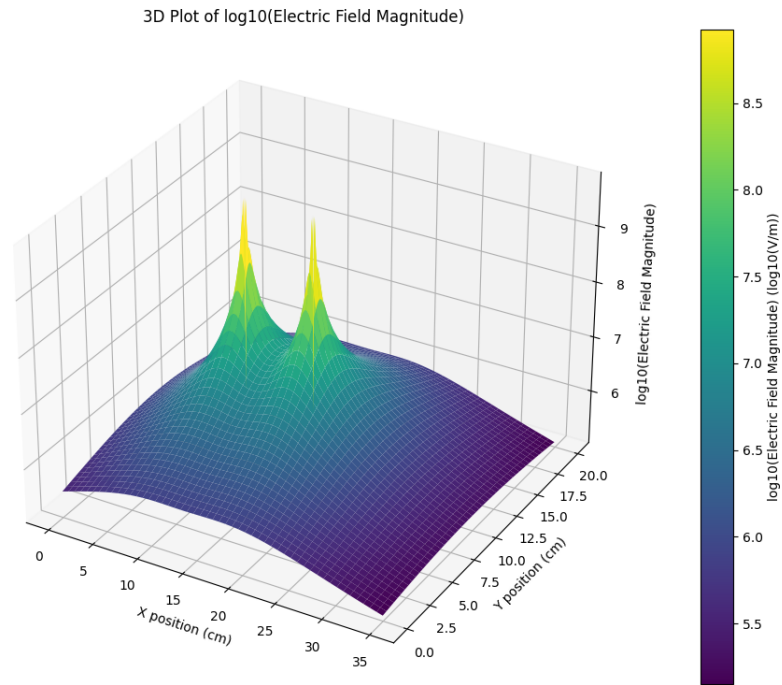
```

1 # Calculate electric field components as negative gradient
2 → of potential
3 Ey, Ex = np.gradient(-V, resolution)
4 # Normalize vector lengths for consistent vector plotting
5 magnitude = np.sqrt(Ex**2 + Ey**2)
6
7 # Re-normalize vector lengths after removing outliers
8 magnitude = np.sqrt(Ex**2 + Ey**2)
9 Ex /= magnitude
10 Ey /= magnitude

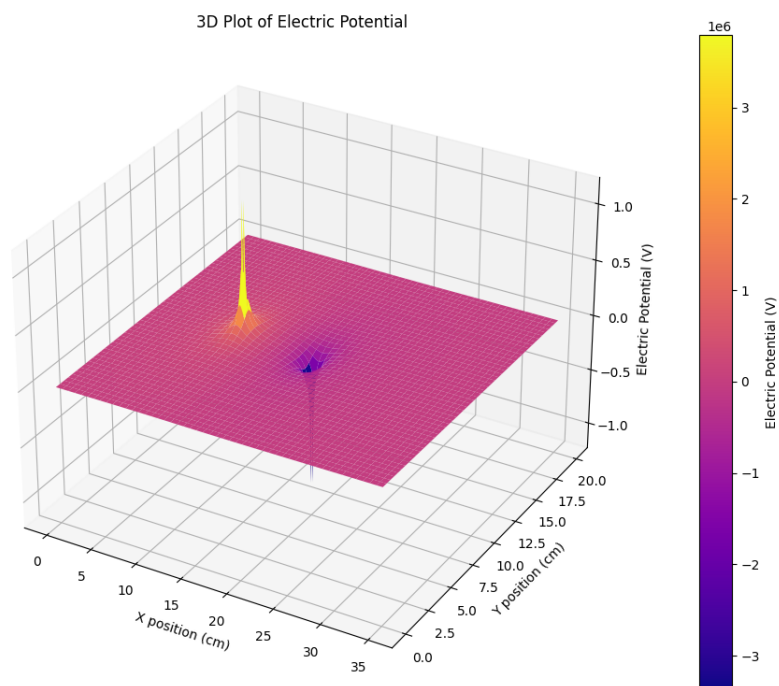
```

$$\log_{10}(|\mathbf{E}|) \quad (7.6)$$

Slične metode za izračun električnog potencijala i polja korištene su u radovima poput Christian Coseru i Plourabou (2021) gdje su analizirani potencijali između točkaste i elektrode većeg promjera, kao i u radu Sohatzadeh i H. Soltani (2021) koji se bavi numeričkom simulacijom električnog polja između gotove jednake geometrije.

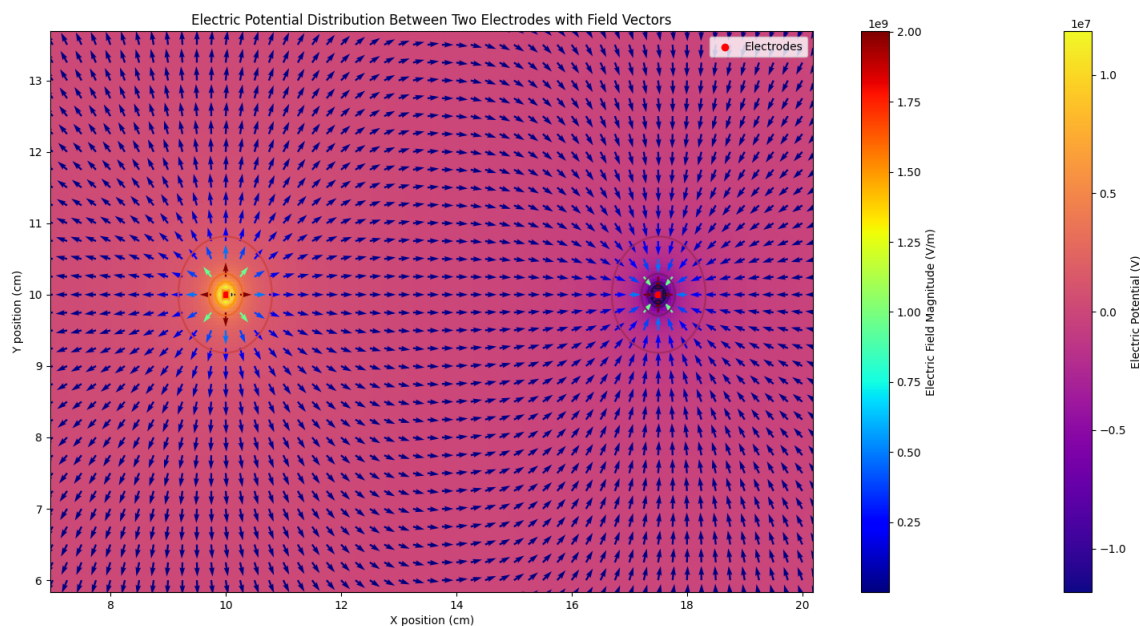


**Slika 7.2:** Prikaz log<sub>10</sub> vrijednosti električnog polja u domeni oko elektroda



**Slika 7.3:** Prikaz vrijednosti električnog potencijala na domeni



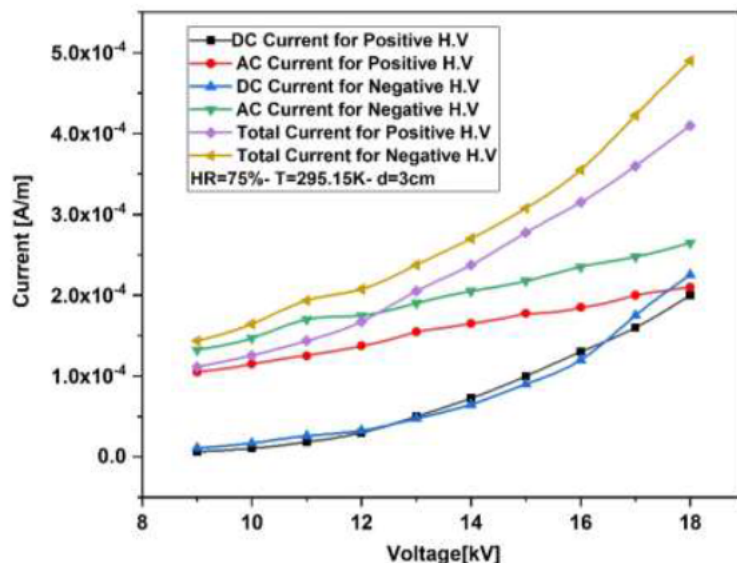


**Slika 7.4:** Prikaz smijera električnog polja na području domene oko elektroda

Ionizacijsko područje kao što je prethodno opisano i definirano u radu Sohbatzadeh i H. Soltani (2021) za ovakav postav djeluje u pretežno pozitivnom  $x$  smjeru. Kako je cilj ovoga rada okupiti kompleksnu spregu između električnog polja, ionizacije, difuzije iona te konačno dinamike fluida u jednostavnu silu koju opisuje ukupno djelovanje svih ovih efekata i utječe direktno na fluid. Numeričke nestabilnosti javljaju se u malim udaljenosti oko elektroda zbog male rezolucije mreže unutar koje je proračun proveden, a radi se o strukturiranoj mreži kojoj je razmak između pojedinih točaka postavljen na 0.5 mm za izvođenje ovih simulacija. Vrijednost u prvim susjednim točkama oko samih elektroda te je zamijenjena srednjom vrijednosti ostalih okolnih točaka udaljenih u smjeru radijalno na većoj udaljenosti od elektrode.

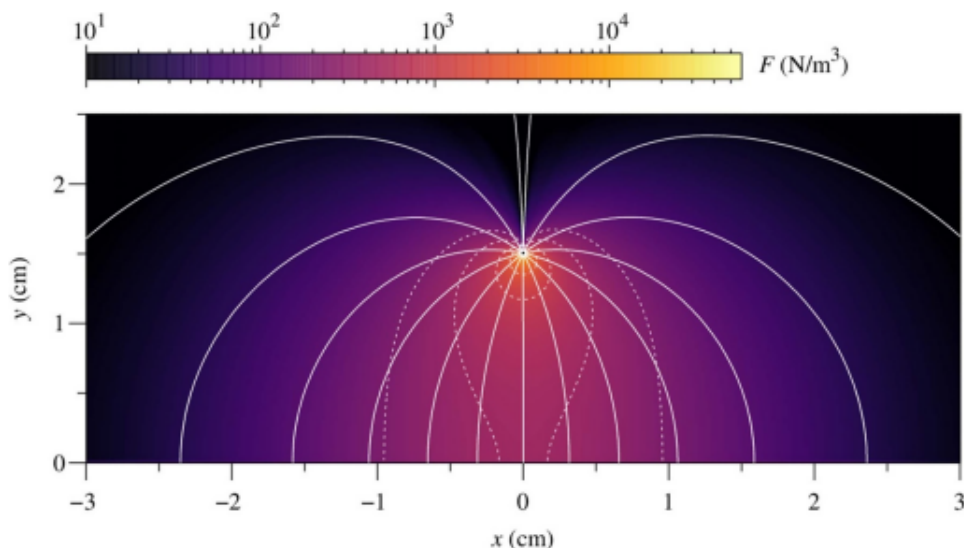
## 7.2.2. Primjena rezultata za spregu s dinamikom fluida

Na temelju gore navedenog postupka određene su izopotencijalne linije kao i smjer djelovanja električnog polja. Uzevši u obzir poznavanje iznosa sile u geometriji u kojoj se koronalni izboj odvija s točke na ravninu moguće je izračunati iznos sile analitički kao što je to prethodno izvedeno 3.32. Aproximacija je provedena tako da se za poznatu udaljenost poznaje otpor zraka te je time određena jakost struje koja prolazi ioniziranim dijelom fluida.



**Slika 7.5:** Iznos struje mjeren na temelju postava iz rada Sohbatzadeh i H. Soltani (2021)

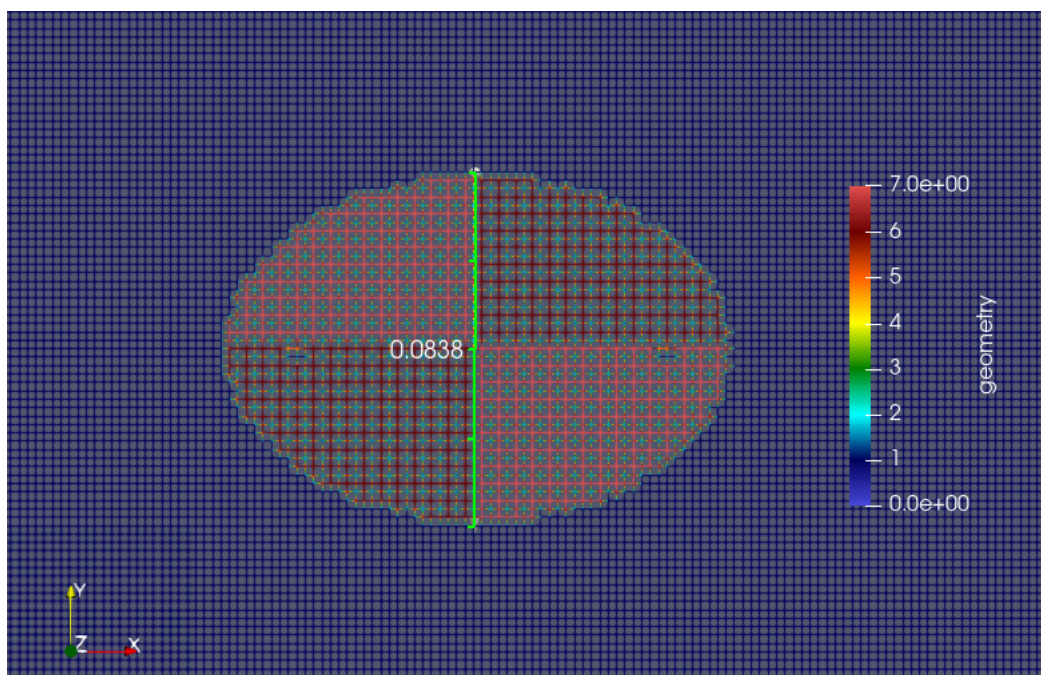
Idući je korak odrediti silu koja se stvara na fluid od struje koja protječe njime. Naime dana analitička funkcija 3.32 prema E.A. i P.S. (1967) određuje silu ravnu plohu sumjerljivu području unutar kojeg dolazi do ionskog vjetra. Takav primjer je prikazan na slici 7.6, a prvotno je prikazan u radu Yanallah (2023).



**Slika 7.6:** Izopotencijala linije (bijelo) i volumna razdioba sile (Yanallah, 2023)

Na temelju prethodno opisane 3.32 moguće je stvoriti vezu između sile koja utječe na plohu kao i sile koju uzrokuje ionski vjetar prema danoj slici 7.6. Vidljiva je poveznica s presjekom područja ionizacije, dakle pretpostavka koja će biti korištena jest da će na određenoj udaljenosti  $d$  od emitterske elektrode sila koju proizvodi struja  $I$  na konstantnoj mobilnosti  $\mu$  sila na fluid biti jednaka kao i na plohu postavljenu na tu udaljenost. Razlika je jedino u

iznosu napona koji je između elektroda. Uslijed toga će na središtu (najvećeg promjera ioni- zacijske regije) sila biti jednaka kao da se na tome mjestu nalazi ploha koja je kolektorska elektroda. Lokacija tako određene plohe prikazana je na slici 7.7.



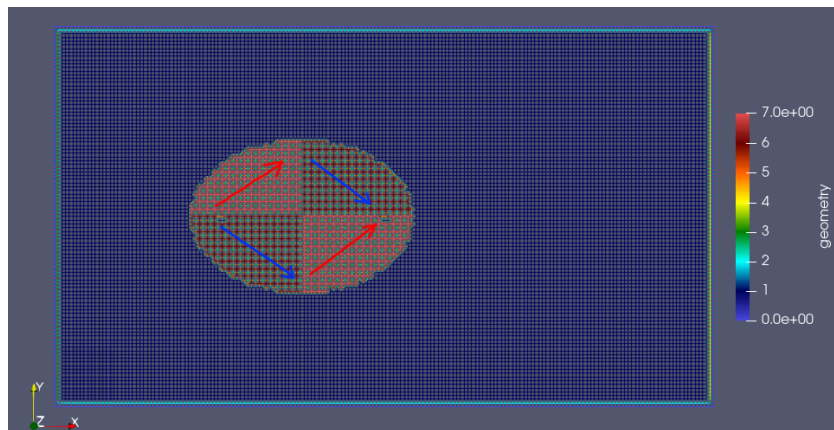
**Slika 7.7:** Linija (zeleno) koja pokazuje lokaciju plohe za izračun očekivanog iznosa sile

Eliptičnim je dijelom na slikama 7.7 i 7.1 prikazano područje djelovanja sile. Ova je aproksimacija postavljena tako da se na temelju izračuna kao u dijelu 7.2.1 te je postavljen iznos koji ograničava dano područje na ono područje unutar kojeg je pri danom naponu i mobilnosti zadovoljen uvjet za stvaranje koronalnog izboja tj. ionizaciju zraka (iznosa reda veličine  $10^6$  V/m) pri danoj mobilnosti na temelju svojstava zraka. Ovakvu je aproksimaciju moguće izvesti pošto opisuje podjednaki prostor domene kao i u radu gdje ona opisuje elipsu na temelju prostorne distribucije iona (Christian Coseru i Plourabou, 2021) te je bolja aproksimacija od pravokutnika prikazanog na slici 6.4 (Nicolas Monrolin i Praud, 2017).

### 7.2.3. Određivanje smjera sile i iznosa

Sila koja djeluje na fluid je prouzrokovana gibanjem iona koji prate smjer djelovanja silnica električnog polja. Dakle, putanje iona koji stvaraju sudare su određene smjerom djelovanja polja, a smjer djelovanja polja je prikazan na slici 6.1. Time je vidljivo kako se do središta puta koje je na slici 7.7 označeno kao referentno područje sile udaljavaju u pozitivnom i negativnom y smjeru ovisno s koje se strane ravnine koje povezuje elektrode nalaze, a zatim u drugom dijelu domene djelovanja sile se približavaju kolektorskoj elektrodi jednakim iznosom i smjerom u x osi dok se smjer u y osi mijenja, a iznos ostaje jednak. Ovim je pristupom

lako za uočiti simetriju. Uzme li se sila u x smjeru konstantnog iznosa, a staze iona moraju početi i završiti u istoj y koordinati samo promijenjene x koordinate lako je podijeliti domene na 4 dijela unutar kojih djeluju dvije različite sile. Ti su materijali na slikama na slici 7.7 i 7.1 označeni kao 6 i 7, a razlikuju se samo u smjeru djelovanja sile, dok je njihova dinamika sudara i toka nepromijenjena odnosno jednaka kao i materijala 1 (fluida). Prikazani su ilustrativni smjerovi djelovanja sile na slici 7.8 te su plavi i crveni vektori istog smjera i orijentacije, a djeluju sukladno prethodno opisnom postupku određivanja smjera djelovanja sile na fluid gibanjem iona.



**Slika 7.8:** Ilustrativni prikaz djelovanja sile u domeni ionizacije

Na materijale nije moguće primijeniti silu koja poprima neku funkciju te je sila uniformno raspoređena na svim dijelovima domene djelovanja. Domena djelovanja sile određena je po proračunu prikazanom u prethodnom poglavlju, a implementacija u kodu koji je korišten za provođenje simulacije i ne uzima u obzir točke unutar domene već označava točke unutar lattice-a na koje je potrebno primijeniti silu kako slijedi:

```

1 // Iterate over x direction (cylinder edge to cylinder
  ↪ edge)
2 for (T x = 0; x <= lengthX; x += latticeSpacing) {
3     for (T y = 0; y <= lengthY; y += latticeSpacing) {
4         bool insideCircle = isPointInsideCircle(x, y,
5             centerCylinderX1, centerCylinderY1,
6             radiusCylinder) ||
7             isPointInsideCircle(x, y,
8                 centerCylinderX2,
9                 centerCylinderY2,
10                radiusCylinder);
11
12         if (!insideCircle) {
13             T E = calculateElectricField(x, y);
14             if (E > thresholdElectricField) {
15                 points.emplace_back(x, y);
16             }
17         }
18     }
19 }

```

Gore implementirane funkcije korištene su kako bi se izbjeglo izračunavanje električnog

polja unutar elektrode (i prostora u kojem se javljaju numeričke nestabilnosti) i radi li se o potencijalu dovoljno visokome za ionizaciju zraka. Implementacija funkcija navedena je u nastavku:

```

1 // Function to calculate electric field magnitude
2 T calculateElectricField(T x, T y) {
3     T distanceToElectrode1 = std::sqrt(std::pow(x -
4         centerCylinderX1, 2) + std::pow(y -
5         centerCylinderY1, 2));
6     T distanceToElectrode2 = std::sqrt(std::pow(x -
7         centerCylinderX2, 2) + std::pow(y -
8         centerCylinderY2, 2));
9     T E = voltageDifference / (distanceToElectrode1 +
10        distanceToElectrode2);
11     return E;
12 }
13 // Function to check if a point is inside a circle
14 bool isPointInsideCircle(T x, T y, T centerX, T centerY, T
    radius) {
15     T dx = x - centerX;
16     T dy = y - centerY;
17     return (dx * dx + dy * dy) <= (radius * radius);
18 }

```

Kao i u cijeloj implementaciji openLB-a korišteni su prototipovi za generički tip podatka  $T$  koji osigurava da jednom definiran tip podataka na početku simulacijskog programskog koda bude konzistentan s ostatkom i pripadajućim funkcijama. Ta je definicija izvedena kao: `using T = FLOATING_POINT_TYPE;`. Sila je određena tako da se postavi virtualna prepreka (ploha) na poziciji označenoj zelenom bojom na slici 7.7 te se izračuna očekivani iznos sile u pozitivnom  $x$  smjeru na tu plohu jer upravo u tom dijelu su svi utjecaji polja u  $y$  smjeru poništeni kako je prikazano na slici 6.1, a uslijed malih brzina i masa iona očekuje se da će utjecaj električnog polja uvelike nadjačati sile koje djeluju na ion. Tako dobivena sila se raspoređuje na točke koje ju stvaraju odnosno točke lattice-a koje stvaraju silu na tu plohu uz udio sile koje djeluje u smjeru pozitivnog i negativnog  $y$  smjera ovisno o dijelu domene. To je sukladno očekivanju da je sila potiska jednaka sili na elektrode kao što je prikazano izvodom koji vodi do jednadžbe 3.32. Pošto je unutar materijala moguće odrediti samo jednu silu koja djeluje na cijeli materijal ta je sila određena tako da je očekivano podijeljena na broj ćelija koje ju stvaraju. Udaljenost je znana na temelju postavljene geometrije, struja je postavljena prema eksperimentalnim podacima kao i napon, a kombinacije koje nisu definirane nije teško predvidjeti s obzirom na cjelokupnost radova u tom području kao što su to Sohbatzadeh i H. Soltani (2021), Christian Coseru i Plourabou (2021) te od velikog značaja i J. Wilso i Thompso (2009). Usporedba između kinetičke energije iona i energije koju električno polje pruža ionima u određenom sustavu. Razumijevanje ove usporedbe je ključno za analizu dinamike gibanja iona u uvjetima ionizirane domene. Kinetička energija iona  $E_k$  može se izračunati pomoću poznate jednadžbe za kinetičku energiju čestice:

$$E_k = \frac{1}{2}m_e v^2 \quad (7.7)$$

gdje je  $m_e$  masa iona, a  $v$  brzina iona. Masa iona, na primjer dušika  $N^+$  označen s  $m_e$  je približno  $2.32499999999991 \times 10^{-26}$  kg.

Električno polje  $\vec{E}$  može utjecati na ion stvarajući potencijalnu energiju  $E_p$  koja se može izračunati pomoću:

$$E_p = e \cdot V \quad (7.8)$$

gdje je  $e$  elementarni naboj ( $e \approx 1.602 \times 10^{-19}$  C), a  $V$  je električni potencijal. Ova jednadžba daje ukupnu energiju koju ion dobiva u prisustvu električnog polja, ali u stvarnosti je potrebno razmotriti i druge faktore kao što su distribucija polja i intenzitet. Za usporedbu kinetičke energije i energije električnog polja potrebno je analizirati odnos između njih. Pretpostavimo da je ion u električnom polju s intenzitetom  $\vec{E}$  i da je u početnom stanju mirovanja. Ako se električno polje uzima kao izvor energije koja povećava kinetičku energiju iona, možemo usporediti promjenu kinetičke energije s potencijalnom energijom stvorenom električnim poljem. Ako je  $E_k$  kinetička energija, a  $E_p$  potencijalna energija, tada se može postaviti odnos:

$$E_k = E_p \quad (7.9)$$

Ako su početni uvjeti takvi da je ion u mirovanju, tada je početna kinetička energija  $E_{k0} = 0$ , a konačna kinetička energija nakon interakcije s električnim poljem bi trebala odgovarati ukupnoj energiji  $E_p$  koju je ion stekao:

$$\frac{1}{2}m_e v^2 = e \cdot V \quad (7.10)$$

Ova jednadžba pokazuje da brzina iona može biti izražena u odnosu na električni potencijal odnosno intenzitet polja te omogućuje lakši uvid u odnose redova veličina u interakcijama između ovih energija te time utjecaja na putanje iona. Uslijed velikog iznosa električnog polja (reda  $10^8$ ) kao i naboja koji je reda  $10^{-19}$  predstavlja mnogo veći iznos energije koji se može razviti u odnosu na kinetičku energiju koja je nekoliko redova veličine manja (brzine reda  $10^1$  i mase reda  $10^{-26}$ ).

Konačno, moguće je pokazati kako je iznos potencijalne energije koju polje razvija reda  $10^{-11} J$  dok je kinetička energija uslijed sudara i usporavanja time, zbog predaje energije na okolni fluid, ograničena na tek  $10^{-25} J$ .

#### 7.2.4. Postavljanje simulacije i uvjeta u domeni

Postavljanje se simulacije odvija u 7 glavnih koraka te će oni biti slijedno opisani za konkretan postav domene i željene ciljeve simulacije. Simulacija je izvedena na temelju pokusnog postava cylinder2d te su mijenjani parametri iste kako bi bilo moguće uobličiti geometriju i postav domene.



## Prvi korak : Inicijalizacija

Inicijalizacija je nužan korak koji određuje parametre za bezdimenzionalan račun koji je prethodno opisan te uz to postavlja objekt `HeuristicLoadBalancer` koji omogućuje dekompoziciju domene. Uz ovo je definiran i broj `cuboid`-a koji opisuju koliko je dijelova domene i kako su definirani za rad na paralelnom proračunu što predstavlja veliku inherentnu prednost ove metode na prethodno opisanim FEM-om i FVM-om.

```
1 // === 1st Step: Initialization ===
2 olbInit( &argc, &argv );
3 singleton::directories().setOutputDir( "./tmp/" );
4 OstreamManager clout( std::cout, "main" );
5
6 UnitConverterFromResolutionAndRelaxationTime const
7   ↪ converter(
8     ↪ int {N}, // resolution: number of
9     ↪ voxels per charPhysL
10    ↪ (T) 0.8, // latticeRelaxationTime:
11    ↪ relaxation time, have to be greater than 0.5!
12    ↪ (T) 2.0*radiusCylinder, // charPhysLength:
13    ↪ reference length of simulation geometry
14    ↪ (T) 1.5, // charPhysVelocity:
15    ↪ maximal/highest expected velocity during simulation
16    ↪ in __m / s__
17    ↪ (T) 0.2*2.*radiusCylinder/Re, // physViscosity:
18    ↪ physical kinematic viscosity in __m^2 / s__
19    ↪ (T) 1.239 // physDensity: physical
20    ↪ density in __kg / m^3__
21 );
22 // Prints the converter log as console output
23 converter.print();
24 // Writes the converter log in a file
25 converter.write("cylinder2d");
26
27 // === 2rd Step: Prepare Geometry ===
28 Vector extend( lengthX, lengthY );
29 Vector origin;
30 IndicatorCuboid2D cuboid( extend, origin );
31
32 // Instantiation of a cuboidGeometry with weights
33 #ifdef PARALLEL_MODE_MPI
34 const int noOfCuboids = singleton::mpi().getSize();
35 #else
36 const int noOfCuboids = 7;
37 #endif
38 CuboidGeometry2D cuboidGeometry( cuboid, L, noOfCuboids );
39 // Instantiation of a loadBalancer
40 HeuristicLoadBalancer loadBalancer( cuboidGeometry );
41 // Instantiation of a superGeometry
42 SuperGeometry superGeometry( cuboidGeometry, loadBalancer
43   ↪ );
```

## Drugi korak : Priprema geometrije

U koraku pripreme geometrije nužno je postaviti veličinu domene kao materijale nužne za izračun. Vidljivo je kako su postavljeni materijali koji su prethodno prikazani i opisani na slici 7.1. Ovaj postupak je ključan za oblikovanje rezultata i točnu konverziju između bezdimenzionalnih brojeva i stvarnih fizičkih veličina.

```

1 void prepareGeometry(UnitConverter const& converter,
2                       SuperGeometry& superGeometry,
3                       std::shared_ptr< circle1,
4                       std::shared_ptr< circle2,
5                       const std::vector>& points)
6 {
7     OstreamManager clout(std::cout, "prepareGeometry");
8     clout << "Prepare Geometry ..." << std::endl;
9
10    Vector extend(lengthX, lengthY);
11    Vector origin;
12
13    superGeometry.rename(0, 2);
14    superGeometry.rename(2, 1, {1, 1});
15
16    // Set material number for inflow
17    extend[0] = 2.*L;
18    origin[0] = -L;
19    IndicatorCuboid2D inflow(extend, origin);
20    superGeometry.rename(2, 3, 1, inflow); //inflow 3, outflow
    ↪ 4
21
22    // Set material number for outflow
23    origin[0] = lengthX-L;
24    IndicatorCuboid2D outflow(extend, origin);
25    superGeometry.rename(2, 4, 1, outflow);
26
27    // Set material number for cylinder
28    superGeometry.rename(1, 5, circle1);
29    superGeometry.rename(1, 5, circle2);
30
31
32    // Assign material 6 and 7 to the specified points
33    for (const auto& point : points) {
34        if ((point[1] > (centerCylinderY1) && ( point[0] >
35            (0.5*(centerCylinderX2 -
36                ↪ centerCylinderX1)+centerCylinderX1))) ||
37            (point[1] < (centerCylinderY1) && ( point[0] <
38                (0.5*(centerCylinderX2 -
39                    ↪ centerCylinderX1)+centerCylinderX1)))) {
40                IndicatorCircle2D forcePoint(point, 0.5 *
41                    ↪ latticeSpacing);
42                // clout << point << std::endl;
43                superGeometry.rename(0, 6, forcePoint);
44                superGeometry.rename(1, 6, forcePoint);
45            }
46            else {
47                IndicatorCircle2D forcePoint(point, 0.5 *
48                    ↪ latticeSpacing);
49                // clout << point << std::endl;
50                superGeometry.rename(0, 7, forcePoint);
51                superGeometry.rename(1, 7, forcePoint);
52            }
53        }
54
55    // Removes all not needed boundary voxels outside the
56    ↪ surface
57    superGeometry.clean();
58    superGeometry.checkForErrors();
59    superGeometry.print();
60
61    clout << "Prepare Geometry ... OK" << std::endl;
62 }

```

### Treći korak : Preiprema lattice-a

Priprema lattice-a je sukladna postupku pripreme mreže (*mesh*) kod FVM i FEM metoda. Definirane su dinamike unutar samog postava te su postavljene granice kao i početni uvjeti simulacije. Dinamika materijala je zadan te je ključan dio primjene funkcije lokalnog ekvi-



librija kako bi se Boltzmannova distribucija 5.1 ispravno primijenila.

```
1 void prepareLattice(SuperLattice& sLattice,  
2                   UnitConverter const& converter,  
3                   SuperGeometry& superGeometry,  
4                   std::shared_ptr< circle1,  
5                   std::shared_ptr< circle2)  
6 {  
7     const T omega =  
8         ↪ converter.getLatticeRelaxationFrequency();  
9     // Material=1 --> bulk dynamics (no-slip boundary  
10    ↪ condition)  
11    auto bulkIndicator =  
12    ↪ superGeometry.getMaterialIndicator({1, 6, 7});  
13    sLattice.defineDynamics(bulkIndicator);  
14    // Material=2 --> bounce back (for solid boundaries)  
15    setBounceBackBoundary(sLattice, superGeometry, 2);  
16    // Set passive boundary conditions  
17    setInterpolatedVelocityBoundary(sLattice, omega,  
18    ↪ superGeometry, 4); // Inflow  
19    setInterpolatedPressureBoundary(sLattice, omega,  
20    ↪ superGeometry, 4); // Outflow  
21    // Material=5 --> bouzidi / bounce back  
22    #ifdef BOUZIDI  
23    setBouzidiBoundary(sLattice, superGeometry, 5,  
24    ↪ *circle1);  
25    setBouzidiBoundary(sLattice, superGeometry, 5,  
26    ↪ *circle2);  
27    #else  
28    setBounceBackBoundary(sLattice, superGeometry, 5);  
29    #endif  
30    // Initial conditions  
31    AnalyticalConst2D rhoF(1);  
32    std::vector velocity(0, T(0));  
33    AnalyticalConst2D uF(velocity);  
34    // Initialize all values of distribution functions to  
35    ↪ their local equilibrium  
36    sLattice.defineRhoU(bulkIndicator, rhoF, uF);  
37    sLattice.iniEquilibrium(bulkIndicator, rhoF, uF);  
38    // Set omega parameter  
39    sLattice.setParameter(omega);  
40    // Initialize lattice  
41    sLattice.initialize();  
42 }
```

```
1 void setBoundaryValues( SuperLattice& sLattice,  
2                       UnitConverter const& converter, int  
3                       ↪ iT,  
4                       SuperGeometry& superGeometry )  
5 {  
6     // No need to set any boundary values since only external  
7     ↪ force is driving the flow  
8 }
```

### Četvrti korak : Petlja i mjerenje vremena

Mjerenje vremena je ključan korak u provođenju simulacije te omogućuje okvirnu vjerodostojnost rezultat, a bitno j jer omogućuje pravilno tumačenje rezultata simulacije. Taj je korak

također neophodan za pravilno formatiranje bezdimenzionalnih brojeva, a u izvornom kodu simulacije je implementiran kao u nastavku.

```
1 // === 4th Step: Main Loop with Timer ===
2 clout << "starting simulation..." << std::endl;
3 clout << "physical time step" << physTimeStep <<
  ↪ std::endl;
4 util::Timer timer( converter.getLatticeTime( maxPhysT ),
  ↪ superGeometry.getStatistics().getNvoxel() );
5 timer.start();
```

### Peti korak : Protok i sudari

Protok i sudari predstavljaju ključ LBM-a te su neizbježan korak u provođenju simulacije. U svakom koraku koji se provodi određena je relaksacijska konstanta odnosno brzina koja opisuje tijek približavanja distribucije koja je trenutna, idealnoj, Lattice-Boltzmannovoj distribuciji 5.1. Kod koji slijedi provodi se u vremenskoj petlji te je jednak i za korake koji slijede te kao takav neće biti ponovljen.

```
1 for (std::size_t iT = 0; iT <
  ↪ converter.getLatticeTime(maxPhysT); ++iT) {
2 // === 5th Step: Collide and Stream Execution ===
3 sLattice.collideAndStream();
4 // === 6th Step: Apply external force
5 applyExternalForceScaled(sLattice, converter,
6 ↪ superGeometry, numCells, points);
7 // === 7th: Computation and Output of the Results ===
8 getResults(sLattice, converter, iT, superGeometry,
9 ↪ timer);
10 }
```

### Šesti korak : Dodavanje sile

Na fluid se u svakom koraku dodaje sila te ovaj pristup omogućuje dinamičku promjenu sile koju stvara električno polje. Silu je nužno primijeniti u svakom koraku kako bi bilo moguće prikazati kontinuirano djelovanje električnog polja na fluid.

### Sedmi korak : Spremanje i obrada rezultata

U ovom je dijelu izražena potreba za converter-om kako bi bilo omogućeno tumačenje podataka koji proizlaze iz simulacije. Moguće je dodati još točaka od interesa unutar domene kao i generiranje slika svakog vremenskog koraka simulacije, no u ovom radu na taj dio mogućnosti nije posvećena veća pažnja pošto su svi podaci kasnije obrađeni korištenjem alata otvorenog koda za vizualizaciju ParaView.

## 7.2.5. Pokretanje simulacije

Uspješno postavljena simulacija pokreće se nakon prevođenja iz programskog jezika C++ u objektni kod koji računalo može izvršiti. Pri uspješnom pokretanju simulacije vidljiv je ispis kao što je prikazan na slici 7.9. Zatim kreće proračun i evaluacija jednadžbi koje opisuju željene fizikalne efekte kao i njihovu spregu. Uspješnost konvergencije moguće je pratiti npr. kontrolnim varijablama, a u ovom je slučaju praćeno tlakom mjerenim netom ispred elektroda te je takav ispis prikazan na slici 7.10.

```
[ThreadPool] Successfully initialized, numThreads=1
[Directories] Directory ./tmp/vtkData/ created.
[Directories] Directory ./tmp/vtkData/data/ created.
[UnitConverter] ----- UnitConverter information -----
[UnitConverter] -- Parameters:
[UnitConverter] Resolution: N= 50
[UnitConverter] Lattice velocity: latticeU= 2.25
[UnitConverter] Lattice relaxation frequency: omega= 1.25
[UnitConverter] Lattice relaxation time: tau= 0.8
[UnitConverter] Characteristical length(m): charL= 0.004
[UnitConverter] Characteristical speed(m/s): charU= 1.5
[UnitConverter] Phys. kinematic viscosity(m^2/s): charNu= 5.33333e-06
[UnitConverter] Phys. density(kg/m^d): charRho= 1.239
[UnitConverter] Characteristical pressure(N/m^2): charPressure= 0
[UnitConverter] Mach number: machNumber= 3.89711
[UnitConverter] Reynolds number: reynoldsNumber= 1125
[UnitConverter] Knudsen number: knudsenNumber= 0.0034641
[UnitConverter] -- Conversion factors:
[UnitConverter] Voxel length(m): physDeltaX= 8e-05
[UnitConverter] Time step(s): physDeltaT= 0.00012
[UnitConverter] Velocity factor(m/s): physVelocity= 0.666667
[UnitConverter] Density factor(kg/m^3): physDensity= 1.239
[UnitConverter] Mass factor(kg): physMass= 6.34368e-13
[UnitConverter] Viscosity factor(m^2/s): physViscosity= 5.33333e-05
[UnitConverter] Force factor(N): physForce= 3.52427e-09
[UnitConverter] Pressure factor(N/m^2): physPressure= 0.550667
[UnitConverter] -----
[prepareGeometry] Prepare Geometry ...
[SuperGeometry2D] cleaned 0 outer boundary voxel(s)
[SuperGeometryStatistics2D] updated
[SuperGeometry2D] the model is correct!
[CuboidGeometry2D] ---Cuboid Structure Statistics---
[CuboidGeometry2D] Number of Cuboids: 7
[CuboidGeometry2D] Delta (min): 0.002
[CuboidGeometry2D] (max): 0.002
[CuboidGeometry2D] Ratio (min): 0.758621
[CuboidGeometry2D] (max): 1.37209
[CuboidGeometry2D] Nodes (min): 2494
[CuboidGeometry2D] (max): 2552
[CuboidGeometry2D] -----
[SuperGeometryStatistics2D] materialNumber=1; count=15772; minPhysR=(0.002,0.002); maxPhysR=(0.348,0.198)
[SuperGeometryStatistics2D] materialNumber=2; count=352; minPhysR=(0,0); maxPhysR=(0.35,0.2)
[SuperGeometryStatistics2D] materialNumber=3; count=99; minPhysR=(0,0.002); maxPhysR=(0,0.198)
[SuperGeometryStatistics2D] materialNumber=4; count=99; minPhysR=(0.35,0.002); maxPhysR=(0.35,0.198)
[SuperGeometryStatistics2D] materialNumber=5; count=4; minPhysR=(0.086,0.1); maxPhysR=(0.176,0.1)
[SuperGeometryStatistics2D] materialNumber=6; count=725; minPhysR=(0.072,0.06); maxPhysR=(0.19,0.14)
[SuperGeometryStatistics2D] materialNumber=7; count=725; minPhysR=(0.072,0.06); maxPhysR=(0.19,0.14)
[prepareGeometry] Prepare Geometry ... OK
```

Slika 7.9: Ispis pri uspješnom pokretanju simulacije

```
[Timer] step=83083; percent=99.7; passedTime=243.358; remTime=0.732274; MLUPs=3.32299
[LatticeStatistics] step=83083; t=9.96996; uMax=0.439558; avEnergy=0.0191379; avRho=0.985358
[getResults] pressure1=0.00476364; pressure2=-0.015582; pressureDrop=0.0203456; drag=inf; lift=-0.0123211
```

Slika 7.10: Ispis pri jednom uspješnom koraku simulacije

## 8. Rezultati

Evaluacija rezultata kod dinamike fluida svodi se na analizu rezultata protoka fluida odnosno na razdiobu tlaka i profil brzina. Uvjeti unutar domene za korištene dvije elektrode (emitterski i kolektorsku) definirani su na sljedeći način:

```
1 // Parameters for the simulation setup
2 const int N = 50; // resolution of the model
3 const T Re = 150.; // Reynolds number
4 const T maxPhysT = 10; // max. simulation time in s, SI
5   → unit
6 const T L = 0.1/N; // latticeL
7 const T lengthX = 0.35;
8 const T lengthY = 0.2;
9 const T centerCylinderX1 = lengthX/4;
10 const T centerCylinderY1 = lengthY/2;
11 const T centerCylinderX2 = lengthX/2;
12 const T centerCylinderY2 = lengthY/2;
13 const T radiusCylinder = 0.002;
14 // Define the physical and lattice parameters explicitly
15 const T charPhysLength = 2.0 * radiusCylinder*10; // Physical
16   → characteristic length
17 const T charPhysVelocity = 0.25; // Physical
18   → characteristic velocity
19 const T physViscosity = 1.8e-5; // Physical
20   → kinematic viscosity
21 const T physDensity = 1.239; // Physical
22   → density
23 const T latticeRelaxationTime = 0.8; // Lattice
24   → relaxation time
25 const T latticeTimeStep = 1.0; // Lattice
26   → time step, typically 1 in LBM
27 // Calculate derived quantities
28 const T latticeSpacing = charPhysLength / N; // Lattice
29   → spacing (delta_x)
30 const T physTimeStep = latticeTimeStep * (latticeSpacing /
31   → charPhysVelocity); // Physical time step
32 // Electric field parameters
33 const T voltageDifference = 12e3; // 12 kV
34 const T distanceBetweenElectrodes = 7e-3; // 7 mm
35 const T thresholdElectricField = 100000; // Threshold
36 // Expected force
37 const T current = 1.6e-6; // in Amps
38 const T ionMobility = 1.8e-4; // in m^2/Vs, example value
39 T force = (current * distanceBetweenElectrodes/2) /
40   → ionMobility; // global force variable
```

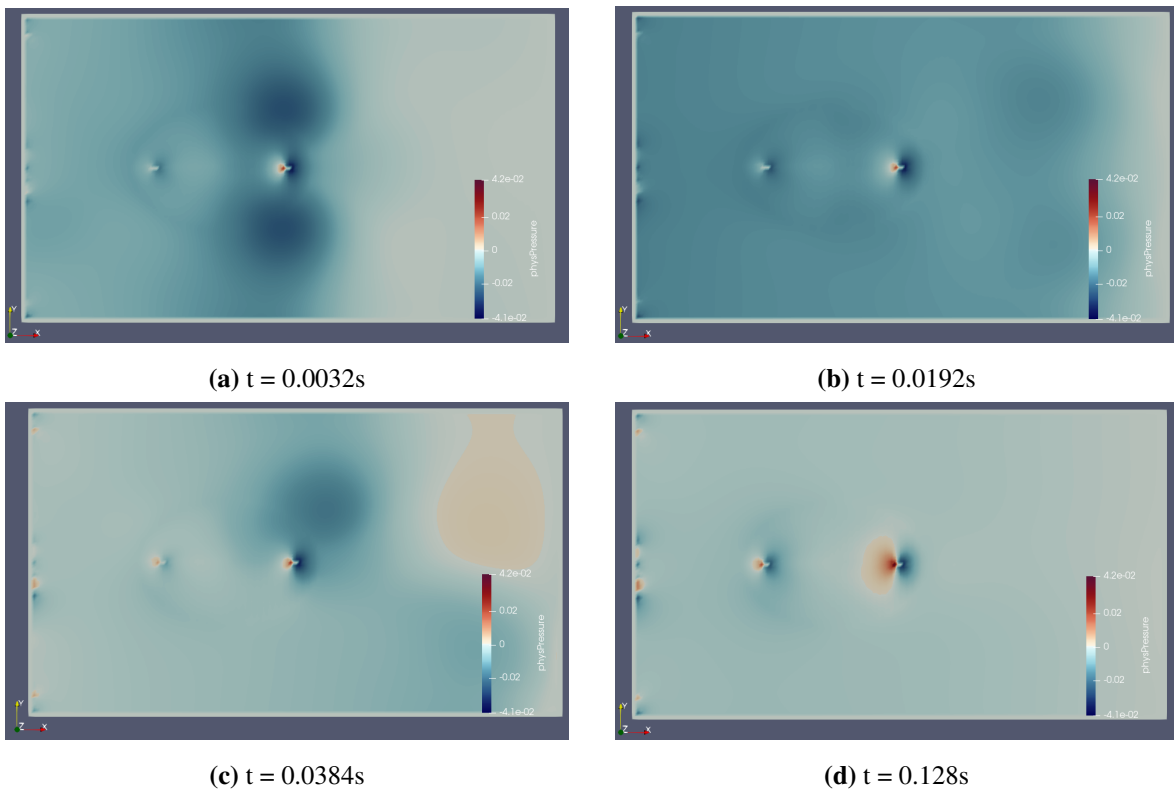
Iako se radi o maloj domeni u relativno niskok razlučivosti u nastavku će biti prikazani tokovi koji pokazuju slično vladanje s mnogo bolje određenim mrežama računanja. Simulacija koja je prethodno prikazana na procesoru Intel i7 G-9750HF, konačno vrijeme koje je potrebno

za izvođenje kao i MLUPs (*Million Lattice Updates Per Second*) prikazan na slici 8.1

```
[Timer] -----Summary:Timer-----
[Timer] measured time (rt) : 244.19s
[Timer] measured time (cpu): 241.564s
[Timer] average MLUPs :      6.070
[Timer] average MLUPps:     6.070
[Timer] -----
```

**Slika 8.1:** Končano vrijeme izvođenja simulacije

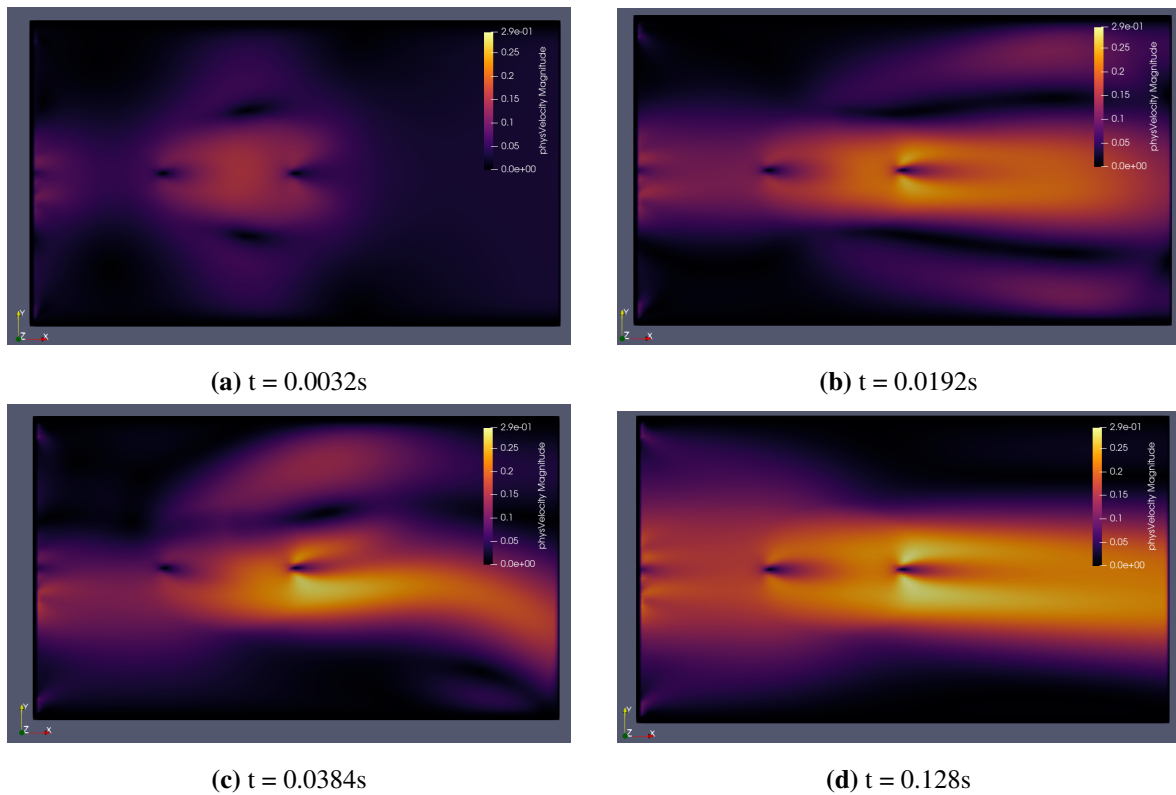
Rezultati simulacije prikazani na idućim slikama, po tlaku 8.2 i po brzini 8.3, razlikuju se u trenutku u kojem su uzeti. Prikazana je geometrija iz prethodnog koda te ju sačinjavaju dvije elektrode.



**Slika 8.2:** Razdioba tlaka po vremenskim trenucima

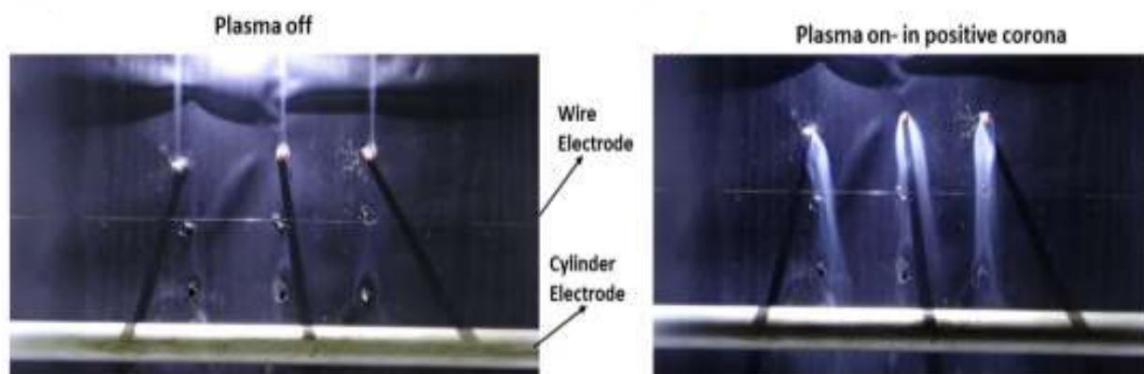
Uslijed malog iznosa viskoziteta zraka prijelazne pojave koje se događaju pri početku simulacije istiravaju se relativno brzo te tok dolazi do svog stabilnog stanja već u vremenu manjem od oko 0.1s. Nakon što tok postane stabilan kao što je prikazano na slici 8.2d kada je razlika tlaka između prednje i stražnje strane kolektorske elektrode mnogo veća u odnosu na razliku ispred emitterske te je na početku domene vidljiva nestabilnost pri stvaranju podtlaka kojeg uzrokuje sila na fluid. Ovaj rezultat nije očekivan i predstavlja numeričku nestabilnost ove implementacije pošto se očekuje neka, ma koliko mala brzina na upuhu. Ovo je rješenje

svejedno uzeto u obzir pošto je domena dovoljno velika da se neželjeni utjecaji ponište (tlak i brzina 8.3d) te se javlja željeni profil toka pri dodiru elektroda tj. brzina i tlak su raspoređeni bez većih odstupanja u razdiobi.



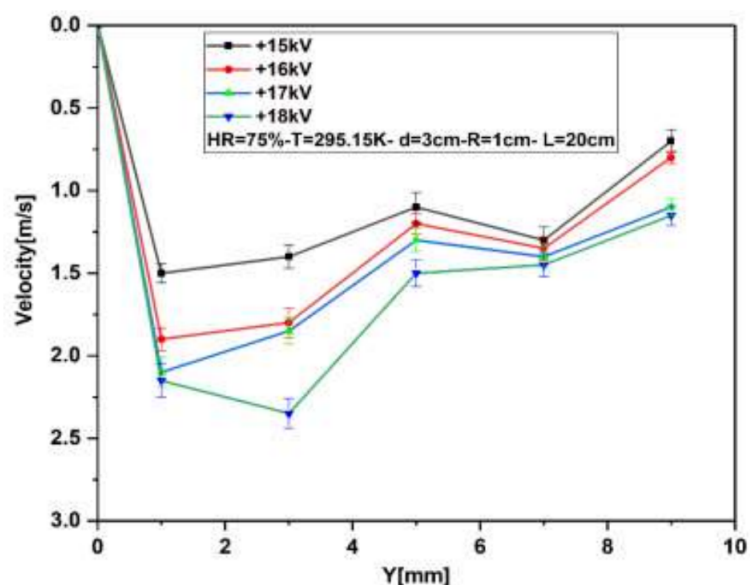
**Slika 8.3:** Profil brzine fluida po vremenskim trenucima

Za usporedbu će biti korišten stvaran postav koji je gotovo u potpunosti sukladan prikazanoj geometriji. Radi se o postavu koji je stvoren i opisan u sklopu rada Sohbatzadeh i H. Soltani (2021). Razlikuju se samo u smjeru djelovanja, odnosno postav je rotiran za  $90^\circ$  u odnosu na simulaciju te je uslijed male mase plina moguće zanemariti utjecaj gravitacije na tako malom području. Dobiveni rezultat prikazan je na slici 8.4 te s razlikom u debljini kolektorske elektrode.

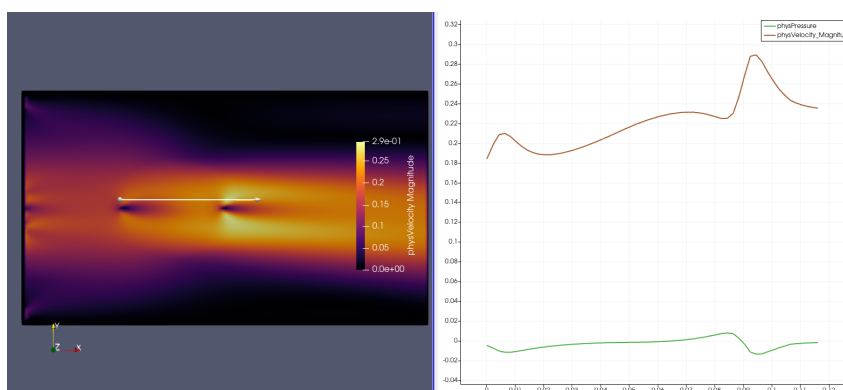


**Slika 8.4:** Prikaz toka oko elektroda pomoću dima (Sohbatzadeh i H. Soltani, 2021)

Lako je uočiti kako dolazi do toka te regije ubravanja, a dobiven potisak je nešto veći u odnosu na analitički proračun (Sohbatzadeh i H. Soltani, 2021). Pri stabilnom toku nakon istitravanja, lako je vidljivo kako se profil brzine toka fluida gotovo u potpunosti poklapa sa simulacijom te je na slici 8.7 prikazano preklapanje dvaju tokova u skali. Radi se o postavu s ponekim promjenama, no moguće je uočiti područje stvaranja potiska (drugu elektrodu) kao i područje niskog tlaka u kojem se dim zadržava. Usporedbom brzina na slici 8.4 na kojoj je izvedeno mjerenje s primjenjenim naponom od 18 kV te prikazuje kako je brzina ovisna o naponu i udaljenosti što je prikazano u radu i na slici 8.5. Brzina i tlak u simulaciji je na samo 12 kV te je očekivana brzina uslijed toga mnogo manja, a iznos brzine i tlaka na liniji prikazanoj bijelom linijom. Skala na grafu za y os m/s, a za x os su metri. Uspoređivanjem danih profila vidljivo je kako dolazi do pada brzine netom iza elektrode te je skok brzine ponovno vidljiv netom prije kolektore elektrode.

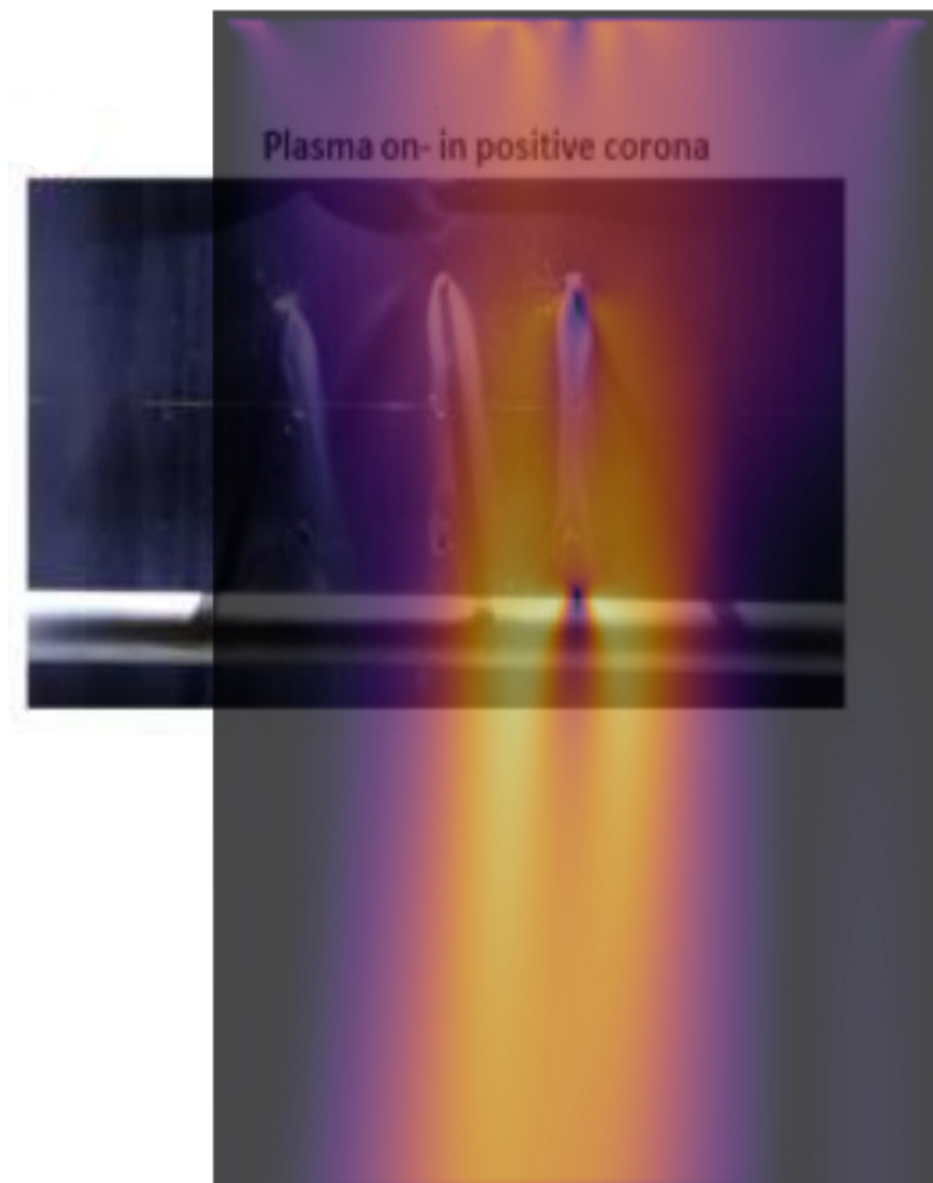


Slika 8.5: Iznos brzine mjeren na postavu ovisno o primijenjenom naponu



Slika 8.6: Iznos brzine u simulaciji uz elektrodu



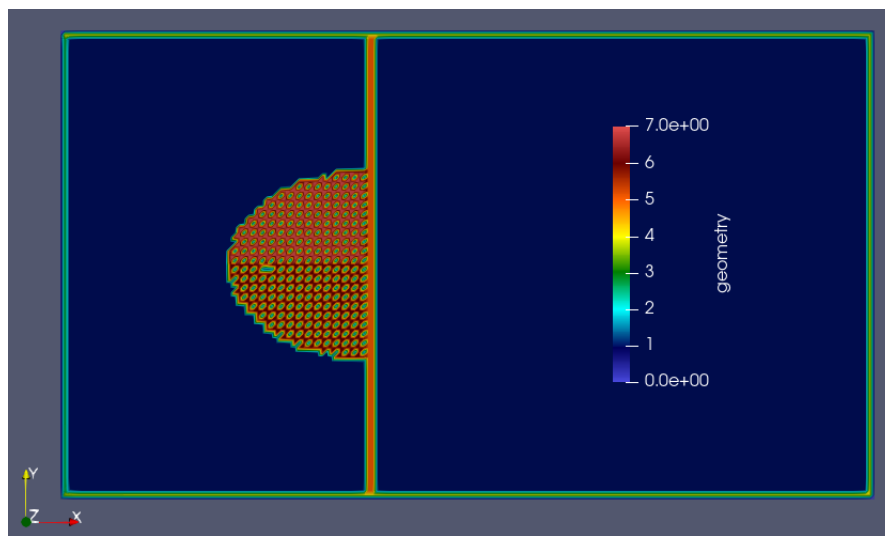


**Slika 8.7:** Usporedba profila brzine simulacije i stvarnog mjerenja dimom

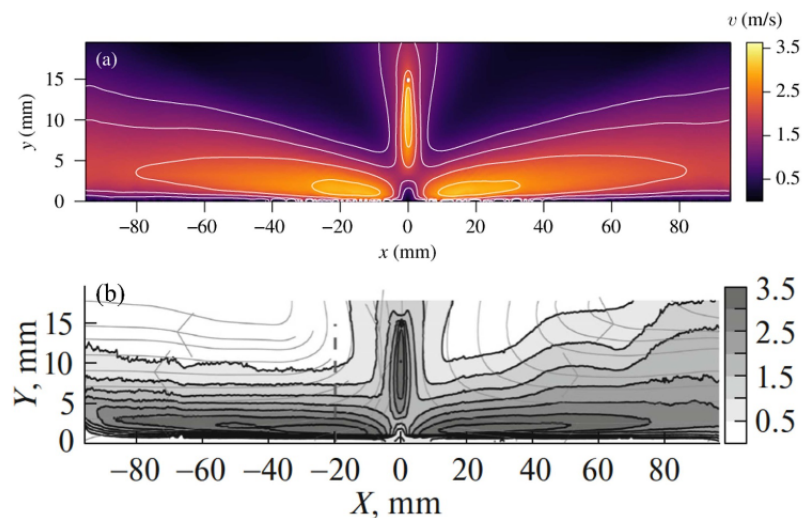
Izvedena je još jedna geometrija koja pobliže opisuje silu koju fluid stvara na plohu prema analitičkom izrazu 3.32. Konačni rezultat takvog proračuna daje nešto pesimističniji iznos sile i brzina kod pozitivnog koronalnog izboja te nešto optimističniji iznos u slučaju negativnog izboja (Sohbatzadeh i H. Soltani, 2021). Uz to omogućuje prikaz koji uspoređuje postav koji sadrži mjerenja Elagin I A i K (2016) i numeričko rješenje prema Yanallah (2023) gdje je dodatno uspoređeno sa stvarnim postavom. Ta geometrija opisuje jednak raspored potencijala i sila kao i prethodna, no razliku čini da ju predstavlja samo jedna elektroda i ploha. Upuh se nalazi na lijevoj strani kao i do sada dok su vrh i dno domene označeni kao ispusi kako ne bi došlo do stvaranja kaverne. Tako izvedena nova geometrija prikazana je na slici



8.8. Sve druge okolnosti postava ostaju nepromijenjene. Rezultati za tlak i brzinu prikazani su sukladno vremenskim intervalima kao i u prethodnom primjeru s dvije elektrode. Slike u početnom trenutku nisu dodane u prikaz pošto se radi samo o jednobožnim pravokutnicima bez ikakvih značajki. Do naglog skok tlaka (vidljiv već u prvom koraku simulacije) dolazi zbog instantnog djelovanja sile tj. sila djeluje na dani dio fluida od prvog trenutka simulacije. Ova implementacija omogućujuće izmjenu iznosa sile tijekom simulacije te otvara mogućnost promjene u svakoj iteraciji simulacije pošto se primjenjuje nakon koraka sudara i protoka. Ostali parametri kao što je regija ionizacije i označavanje materijala tj. postavljanje uvjeta unutar domene nije moguće dinamički mijenjati pošto se taj korak izvodi u inicijalizaciji simulacije.



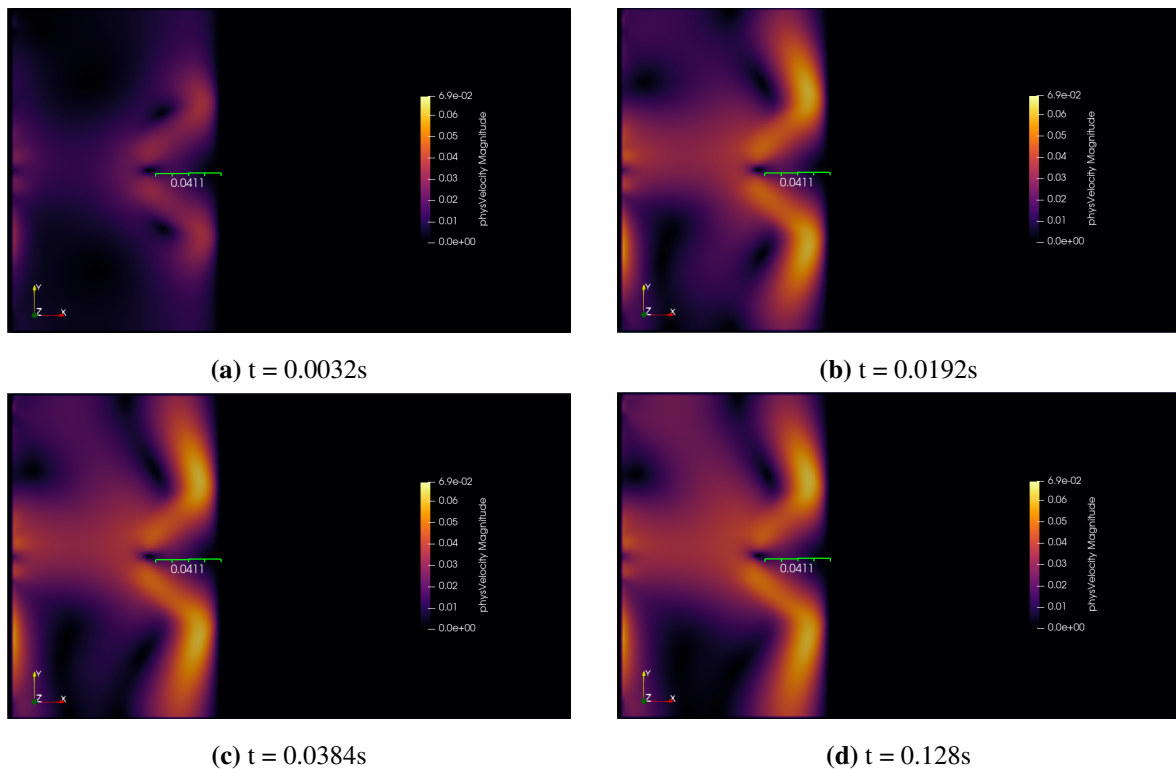
**Slika 8.8:** Geometrija i domena s jednom elektrodom i plohom



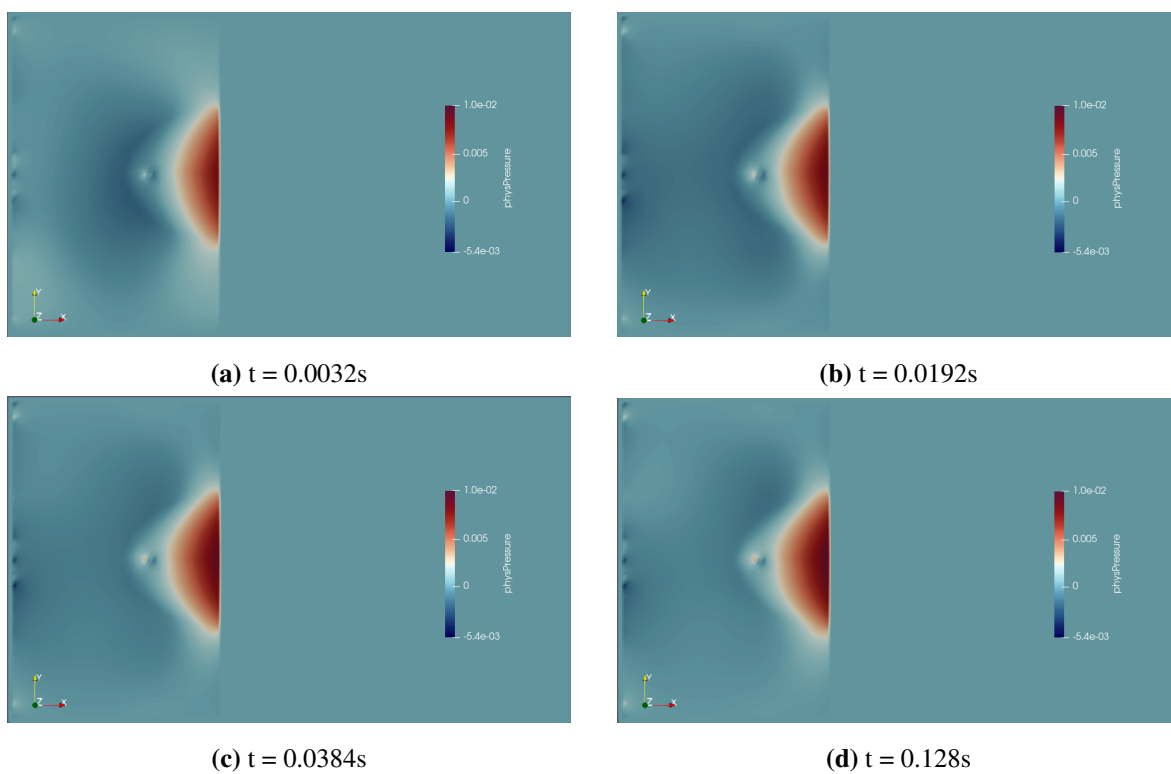
**Figure 7.** 2D-spatial distribution of the velocity magnitude and their corresponding contour lines for an applied voltage  $V = +13.5$  kV (wire radius:  $r_0 = 50$   $\mu\text{m}$ , wire-plate separation:  $d = 15$  mm). (a) Numerical modeling. Isolines of the velocity magnitude are drawn at 2.8, 2.1, 1.5 and 1  $\text{m s}^{-1}$ . (b) Experimental measurements using PIV. Reproduced from [38], with permission from Springer Nature.

**Slika 8.9:** Usporedba openFOAM simulacije i stvarnih mjerenja prema (Yanallah, 2023)

Na danim prikazima može se primijetiti kako dolazi do područja izrazito male brzine (gotovo 0) uz plohu ispod elektrode te se javljaju dva toka prema ispusima s oba ruba plohe. Dani su rezultati uspoređeni sa stvarnim mjerenjem te je vidljiv mali disbalans između danih tokova (lijevi i desni tok nisu potpuno zrcalni). Duljine o kojima se radi između postava nisu potpuno sukladne no vidljivo je kako se radi o dvije zone bržeg fluida te zoni niskog tlaka ispod elektrode. Širine ploha su u oba slučaja jednake te je vidljivo kako iz upuha dolazi zrak. Također na usporedbi simulacija vidljivo je istitravanje te se također javlja disbalans između dviju strana toka kao i na mjenom postavu. Razmak elektroda na slici 8.9 (Yanallah, 2023) iznosi 0.015 m dok je openLB simulaciji taj razmak veći s 0.0411 m, a razlika u primijenjenom naponu je 1.5kV te uslijed toga nije ni za očekivati potpuno jednaku dinamiku postava. Animacije koje predstavljaju izlaz simulacije moguće je preuzeti na vezi.



**Slika 8.10:** Profil brzine fluida po vremenskim trenutcima u geometriji točke i plohe



**Slika 8.11:** Razdioba tlaka po vremenskim trenutcima geometriji točke i plohe

## 9. Zaključak

U radu je prikazana primjena LBM-a koja pruža veoma visoku efikasnost i brzo izvođenje numeričkih simulacija vezanih za tok fluida uz inherentnu sposobnost za paralelno računanje. Zbog takvih je prednosti izabrana na FEM i FVM metodama te je korištena openLB biblioteka koja implementira potrebne metode i sadrži probne primjere koje je moguće modificirati kako bi bilo lakše dobiti željena vladanja.

Izvedeni su osnovni zakoni dinamike gibanja fluida pod utjecajem električnog polja sprejom sudara ioniziranog dijela plina. Ovakav je pristup omogućio implementaciju koja daje relativno jednostavan način analitičkog proračuna sile i dobivanje rješenja simulacije koja uz malo računalnih resursa ostvaruje dobru aproksimaciju toka.

Izvedene su dvije geometrije koje predstavljaju poznate probleme postave u mjerenjima i simuliranjima vladanja zraka uslijed djelovanja ionskog vjetra te su uspoređeni profili brzina s već ostvarenim kompleksnijim simulacijama i stvarnim mjerenjima na izvedenim postavima. Pokazalo se kako su dobiveni rezultati oblikom slični očekivanim rezultatima te su iznosi brzina slični i sumjerljivi što je za očekivati s obzirom na veliki broj aproksimacija i pojednostavljenja koja su izvedena u sklopu ovog rada, no i dalje se zadržava sposobnost realnog prezentiranja stvarnog svijeta.

Ova metoda omogućuje brz i kompaktan način za prikaz toka fluida i vjerno opisuje djelovanje u stvarnom svijetu. Moguće je izvesti još neka poboljšanja što se tiče odabira parametara, no za to je potrebno korištenje metode koja uvodi manje aproksimacija (pune modele difuzije i sudara) kao i izgradnju mjernog postava te provođenje mjerenja koja su u potpunosti jednakih parametara kao i simulacije.

# LITERATURA

- K. J. Bathe. Finite element procedures. *Prentice Hall*, 1996.
- J. Blazek. *Computational Fluid Dynamics: Principles and Applications*. Elsevier, 2001.
- J. R. Clausen C. K. Aidun. Lattice-boltzmann method for complex flows. *Annual Review of Fluid Mechanics*, 2010.
- David Fabre Christian Coseru i Franck Plourabou. Numerical study of electroaerodynamic force and current resulting from ionic wind in emitter/collector systems. *Journal of Applied Physics* 129, 2021.
- R. D. Cook, D. S. Malkus, i R. J. Witt M. E. Plesha. Concepts and applications of finite element analysis. *John Wiley and Sons*, 2001.
- Christenson E.A. i Moller P.S. Ion-neutral propulsion in atmospheric media. *AIAA Journal* vol. 5, no. 10, pp. 1768–1773, 1967.
- Ashikhmin I A Elagin I A, Yakovlev V V i Stishkov Y K. Experimental investigation of cooling of a plate by ionic wind from a corona-forming wire electrode. *Tech. Phys.* 61 1214–9, 2016.
- J. Jiménez F. J. Higuera. Boltzmann approach to lattice gas simulations. *Europhysics Letters*, 1988.
- U. Frisch i Y. Pomeau B. Hasslacher. Lattice-gas automata for the navier-stokes equation. *Physical Review Letters*, 1986.
- G. Zanetti G. R. McNamara. Use of the boltzmann equation to simulate lattice-gas automata. *Physical Review Letters*, 1988.
- W. Malalasekera H. K. Versteeg. An introduction to computational fluid dynamics: The finite volume method. *Pearson Education Limited*, 2007.
- T. J. R. Hughes. *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*. Dover Publications, 1987.

- M. Peric J. H. Ferziger. Computational methods for fluid dynamics. *Springer*, 2002.
- Hugh D. Perkins J. Wilso i William K. Thompso. An investigation of ionic wind propulsion. *NASA*, 2009.
- H. Jasak i Z. Tukovic A. Jemcov. Openfoam: A c++ library for complex physics simulations. *International Workshop on Coupled Methods in Numerical Dynamics, IUC Dubrovnik*, 2007.
- N. A. Kaptsov. Elektricheskie yavleniya v gazakz i vakuume. *OGIZ, Moscow*, 1947.
- M. J. Krause, A. Kummerländer, S. J. Avis, H. Kusumaatmaja, D. Dapelo, F. Klemens, M. Gaedtke, N. Hafen, A. Mink, R. Trunk, J. E. Marquardt, M.-L. Maier, M. Haussmann, i S. Simonis. Openlb—open source lattice boltzmann code. *Computers & Mathematics with Applications*, 2020. doi: 10.1016/j.camwa.2020.04.033.
- T. Kruger i A. Kuzmin H. Kusumaatmaja. *The Lattice Boltzmann Method: Principles and Practice*. Springer, 2017.
- M. Krafczyk L. Hegele. Openlb – open source lattice boltzmann code. *OpenLB*, 2018.
- P. Råback M. Malinen. Elmer finite element software. *scs.fi*, 2008.
- Jonathan B. Jeppener-Haltenhoff Marc Haussmann, Marc Haussmann, Yongxiang Li, Marius Schmidt, Cooper Welch, Lars Illmann, Benjamin Böhm, Hermann Nirschl, Mathias J. Krause, i Amsini Sadiki. Evaluation of a near-wall-modeled large eddy lattice boltzmann method for the analysis of complex flows relevant to ic engines. *MDPI*, 2020.
- A.K.M. Monayem H. Mazumder i F.C. Lai. Performance of a multiple stage ehd gas pump in a square channel. *ASME 2011 International Mechanical Engineering Congress and Exposition (277-284)*, 2011.
- A.K.M.M.H. Mazumder, X.B. Zhao, i F.C. Lai. Effects of grounded electrodes size on the performance of ehd gas pump in a square channel. *Proc. ESA Annual Meeting on Electrostatics*, 2011.
- F. Moukalled i M. Darwish L. Mangani. The finite volume method in computational fluid dynamics: An advanced introduction with openfoam and matlab. *Springer*, 2016.
- Franck Plourabou ´e Nicolas Monrolin i Olivier Praud. Electrohydrodynamic thrust for in-atmosphere propulsion. *AIAA*, 2017.
- R. L. Taylor O. C. Zienkiewicz. The finite element method: Its basis and fundamentals. *Butterworth-Heinemann*, 2005.

- F. W. Peek. Dielectric phenomena in high voltage engineering. *McGraw Hill, New York*, 1929.
- G. Doolen S. Chen. Lattice boltzmann method for fluid flows. *Annual Review of Fluid Mechanics*, 1998.
- E. Rank S. Kollmannsberger. Parallel computing in computational fluid dynamics with lattice boltzmann methods. *Springer*, 2017.
- R.S. Sigmond. Simple approximate treatment of unipolar space-charge-dominated coronas: The warburg law and the saturation current. *Journal of Applied Physics*, vol. 53, 1982.
- S. Simonis. Lattice boltzmann methods for partial differential equations. *Doctoral thesis. Karlsruhe Institute of Technology (KIT)*, 2023.
- F. Sohbatzadeh i S. Mirzanejhad H. Soltani. The role of nonlinear body force in electrostatic propeller driven by atmospheric dc corona dischargesn. *Journal of Electrostatics*, vol. 110, pp. 103-113, 2021.
- S. Succi. The lattice boltzmann equation for fluid dynamics and beyond. *Oxford University Press*, 2001.
- S. Succi. *A Practical Introduction to the Lattice Boltzmann Method*. Oxford University Press, 2008.
- Martin Seeger Thomas Christen. Current interruption limit and resistance of the self-similar electric arc. *Journal of Physics D: Applied Physics*, 2005.
- J.S. Townsend. Electricity in gases. *Clarendon Press*, 1915.
- J.S. Townsend i John Sealy Edward. The theory of ionization of gases by collision. *London, Constable*, 1910.
- A.M. Tyndall i G.C. Grindley. The mobility of ions in air. part i. negative ions in moist air. *Proceedings of the Royal Society (London)*, vol. 110, no. 754, pp. 341–358, 1926.
- Alexander J. Wagner. *A Practical Introduction to the Lattice Boltzmann Method*. Fargo, 2008.
- K. Yanallah. A new numerical approach for efficient modeling of positive corona discharge and its associated electric wind. *Phys. D: Appl. Phys.* 56 415201, 2023.

## Simulacija ionskog vjetra temeljem Lattice-Boltzmannove metode

### Sažetak

Čvrsta elektrostatička pogonska tehnologija predstavlja značajan napredak u području svemirske propulzije, omogućujući kretanje u vakuumu korištenjem električki nabijenih čestica za generiranje potiska. Ovaj rad istražuje mogućnost primjene ionskih potisnika, specifične vrste čvrstih elektrostatičkih pogona, u atmosferskim uvjetima. Ionski potisnici, koji su se tradicionalno koristili u svemiru, sada pokazuju potencijal za primjenu unutar atmosfere, zahvaljujući nedavnim tehnološkim napretcima.

U radu se detaljno razmatra princip rada ionskih potisnika i ključna uloga koronalnog izboja u generiranju potiska bez pokretnih dijelova. Simulacijski slučajevi provedeni su korištenjem metode Lattice-Boltzmann (LBM) koja omogućava brzo izvođenje numeričkih simulacija i paralelno računanje, čime se povećava efikasnost analize fluidnih tokova uzrokovanih ionskim vjetrom.

Kao dio istraživanja, provedene su simulacije na dvjema različitim geometrijama, a rezultati su uspoređeni s postojećim simulacijama i eksperimentalnim mjerenjima. Dobiveni rezultati pokazali su sličnosti u obliku i veličini brzinskih profila, što ukazuje na dobru aproksimaciju stvarnih fizikalnih pojava uz korištenje relativno jednostavne numeričke metode.

Zaključeno je da LBM metoda pruža učinkovito sredstvo za simulaciju i analizu elektrostatičkog pogona u atmosferi, iako su potrebna daljnja istraživanja za poboljšanje modela, smanjenje aproksimacija i provedbu dodatnih eksperimentalnih mjerenja koja bi omogućila preciznije validiranje simulacijskih rezultata.

**Ključne riječi:** ionski vjetar, Lattice-Boltzmannova metoda, ionska propulzija, CFD



## Abstract

Solid electrostatic propulsion technology represents a significant advancement in space propulsion, enabling movement in a vacuum by utilizing electrically charged particles to generate thrust. This paper explores the potential application of ion thrusters, a specific type of solid electrostatic propulsion, in atmospheric conditions. Ion thrusters, traditionally used in space, now show potential for atmospheric applications due to recent technological advancements.

The paper provides an in-depth examination of the operating principles of ion thrusters and the critical role of corona discharge in generating thrust without moving parts. Simulation cases were conducted using the Lattice-Boltzmann Method (LBM), which allows for fast numerical simulations and parallel computing, thereby enhancing the efficiency of analyzing fluid flows induced by ionic wind.

As part of the research, simulations were performed on two different geometries, and the results were compared with existing simulations and experimental measurements. The obtained results showed similarities in the shape and magnitude of the velocity profiles, indicating a good approximation of real physical phenomena using a relatively simple numerical method.

It is concluded that the LBM method provides an effective means for simulating and analyzing electrostatic propulsion in the atmosphere, although further research is needed to improve the models, reduce approximations, and conduct additional experimental measurements that would enable more precise validation of the simulation results.

**Keywords:** ionic wind, Lattice-Boltzmann method, ionic propulsion, CFD