

Универзитет у Београду
Физички факултет

Нумеричке симулације процеса хлађења грануларног гаса

-Мастер рад-

Владан Симић

2015/7018

Београд

Септембар, 2017.

Ментор: др Слободан Врховац, научни саветник
Институт за физику, Београд

Чланови комисије:

др Слободан Врховац, научни саветник
Институт за физику, Београд

проф. др Андријана Жекић, ванредни професор
Физички факултет, Универзитет у Београду

др Зорица Поповић, доцент
Физички факултет, Универзитет у Београду

Датум одбране:

Захваљујем се свом ментору, Слободану Врховцу, научном саветнику у Институту за физику у Београду на свесрдној помоћи при изради овог мастер рада.

Такође се захваљујем Дијани Дујак, научном сараднику на Факултету за металургију и метале, Универзитета у Зеници и Александру Карачу, редовном професору на Политехничком факултету, Универзитета у Зеници на уступању прелиминарних верзија програмских кодова којима је моделована динамика грануларног гаса.

Садржај

Први део – Теорија грануларних гасова

1	Увод	6
2	Грануларни материјали	9
2.1	Особине и подела	9
2.2	Основне особине грануларних гасова	11
2.3	Слободно хлађење грануларног гаса	12
3	Динамика судара и <i>event-driven</i> алгоритам	13
3.1	Судари честица	14
3.2	Нацрт <i>event-driven</i> алгоритма	17
4	Реализација кода на основу <i>event-driven</i> алгоритма	19
4.1	Кретање честица по инерцији између судара	19
4.2	Сударање две честице	19
4.3	Судари са зидовима	22
4.4	Периодични гранични услови	23
4.5	Распоређивање почетних положаја и брзина	23
4.6	Утврђивање распореда судара	24
4.7	Главни програм	25
4.8	Прикупљање и обрада података	27
5	Појава нееластичног колапса у нумеричким симулацијама	28
5.1	Коефицијент реституције	29
5.2	Методe решавања нееластичног колапса	31
6	Кинетичка теорија грануларних гасова и Хафов закон	32
6.1	Нова временска скала	33

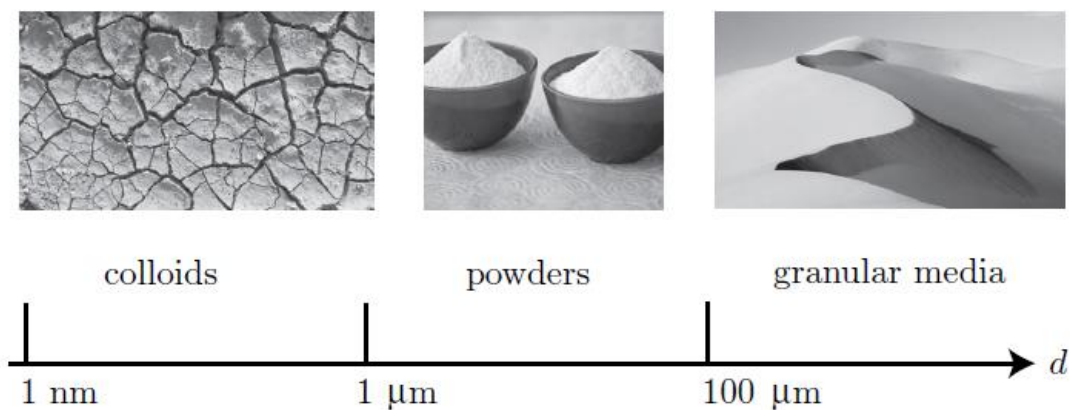
Други део - Презентовање и анализа резултата

7	Поређење различитих метода избегавања нееластичног колапса у нумеричким симулацијама	36
8	Динамика хлађења грануларног гаса у две димензије	40
8.1	Кинетичка фаза	41
8.2	Фаза смицања	45
8.3	Фаза кластерована	51
8.4	Периодични гранични услови	53
9	Закључак	60
10	Литература	61

Први део – Теорија грануларних гасова

1 Увод

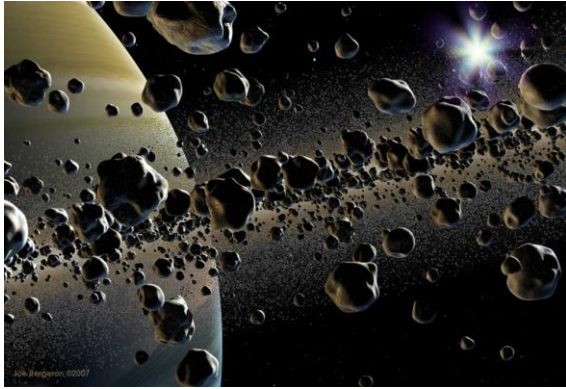
Грануларни гасови су део веће групе материјала које називамо грануларним. Не постоји прецизна дефиниција грануларних система, међутим можемо рећи да су то системи великог броја макроскопских честица са пречником већим од $100\ \mu\text{m}$ (слика 1.1). При томе се подразумева да се честице не распадају током кретања [1]. Поред тога, силе које делују између честица су одбојне, и делују на граници честица. Такође, могу бити укључене и спољашне силе, нпр. гравитација. Основно својство грануларних система је нееластичност честица приликом судара. Услед нееластичности долази до дисипације кинетичке енергије честица и њихово претварање у топлотну или енергију звучних таласа.



Слика 1.1 – скала различитих материјала

Грануларне материјале можемо сусрести свуда у природи, од песка, преко зрна шећера и соли, до великих система као што су планинске лавине, или чак астрономских као што су нпр. Сатурнови прстенови (слика 1.2). Многи аутори сматрају да их због својих карактеристичних својстава не можемо сврстати у неке од постојећих агрегатних стања (иако су они појединачно у чврстом стању), већ кажу да они формирају ново агрегатно стање [2].

Потреба за научним разумевањем грануларних система је велика. Разумљиво је зашто је оно потребно код изучавања лавина, међутим своју примену наука о грануларним системима налази и у агрикултури (при складиштењу и транспорту житарица), фармацији, као и у енергетици, при транспорту угља, на пример, који се преноси помешан са нафтом. У последњем случају бисмо имали посао са „грануларним течностима“. Процењује се да су тренутни губици, услед непотпуног разумевања динамике грануларних материјала, око 40% [2]. Паметним улагањем у системе за транспорт и проучавање истих, могли бисмо уштедети велика средства у тим областима.



Слика 1.2 – примери грануларних материјала

Садашње стање у проучавању грануларних материјала није на високом нивоу разумевања као неке друге гране физике. То важи чак и за најједноставније системе које ћемо ми проучавати у овом раду, систем идентичних сферних честица. То може звучати као парадокс, јер је физика која описује ове системе класична, Њутнова механика. Следећи разлози могу бити разлог томе:

- **Велики број честица**

Ако бисмо посматрали само 1 cm^3 шећера, узимајући да је величина зрна $100 \mu\text{m}$, добијамо да је број зрна, уз коефицијент компактности 0,5, приближно једнак $\frac{10^{-6} \text{ m}^3}{2 \cdot 10^{-12} \text{ m}^3} \cdot 0.5 = 250\,000$. Овај број представља проблем и за најсавременије рачунаре на којима би се вршила симулација. Међутим, и у термодинамици постоји тај проблем, па опет можемо да добијемо ваљану теорију. Зашто? Следећи разлог нам даје одговор.

- **Термалне флуктуације су занемарљиве**

Код „обичних“ течности и гасова термално кретање омогућава молекулима да брзо достигну равнотежни положај, где, ако имамо ансамбл таквих система, сваки равнотежни положај има одговарајућу статистичку тежину и на основу микроскопских параметара можемо увести макроскопске величине. То није случај са грануларним материјалима. Честице су доста велике да би имале Брауново кретање и статистичко усредњавање је готово немогуће. Грануларни материјали су према томе атермални системи [1]. Да бисмо ово илустровали можемо посматрати честице стакла величине 1 mm , густине $2500 \frac{\text{kg}}{\text{m}^3}$, на собној

температури и у гравитационом пољу Земље. Гравитациона енергија, која кореспондира термалној у статистичкој механици, би била $E_p \sim mgh = 10^{-10} J$. Док би термална била $k_b * T = 4 * 10^{-21} J$. Можемо, на основу масе честица проценити пречник честица на којима би термална енергија имала значајнију улогу и он износи $\approx 1 \mu m$.

- **Не постоје разлике у реду величине честица и њихове макроскопске скале кретања**

На пример, када песак пада на гомилу у пешчаном сату, ширина тока је свега двадесетак пута већа од честица, тако да се доводи у питање могућност увођења хипотезе континуума.

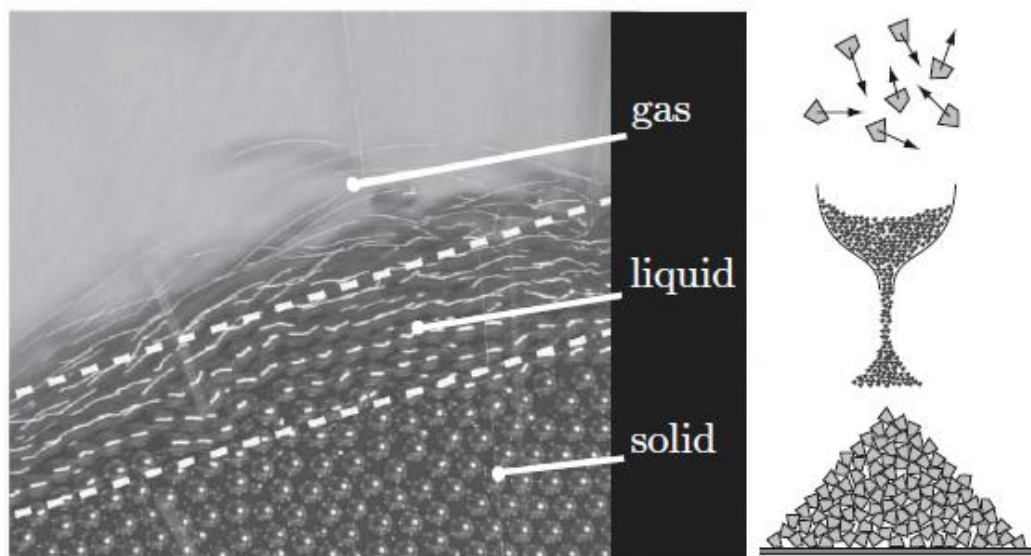
- **Интеракције између честица могу бити сложене**

Дисипација енергије може на веома сложен начин да зависи од релативних брзина честица приликом судара. Осим тога, силе трења су увек присутне и такође на сложен начин утичу на динамику система.

2 Грануларни материјали

2.1 Особине и подела

Грануларне материјале можемо поделити на чврсте, течне и гасовите (слика 1.3), углавном на основу стања кретања у којем се налазе, као и густине система честица¹.



Слика 1.3

Ако је средња енергија кретања мала и ако су честице приближно у стању мировања једна према другој, можемо рећи да имамо чврсто грануларно стање. На пример, ако имамо гомилу песка у стању мировања, под углом мањим од критичног угла покретања² имамо такво стање. Међутим, ако бисмо повећали тај угао, по површини би уочили кретање које подећа на кретање течности. Наравно, то кретање се веома разликује од протока течности. Битна разлика је, на пример, што се одвија само по површини, као што можемо видети на слици 1.3.

Грануларни материјали се одликују разним необичним особинама. Једна од њих је да ако посматрамо гомилу таквих честица (нпр. песка, житарица) у неком силосу, мерећи притисак на дно силоса долазимо до закључка да он расте до неке висине, а потом остаје константан. То је последица сила трења међу честицама и између честица и зидова који због тога трпе већи притисак. Из истог разлога песак у пешчаном сату протиче приближно константно.

¹ Под густином система честица се подразумева запремина свих честица подељена запремином система

² Repose angle (енг.)

Друга особина грануларних материјала је начин паковања честица. Насумично паковање сферних честица први је проучавао Стефан Хејлс (Stephen Hales) у 18. веку. Показано је да, зависно од начина попуњавања суда, скуп честица може да буде смештен у њему са коефицијентом компактности (запремина свих честица по запремини суда) између 0,55 до 0,64. Због статичке силе трења, честице могу остати у метастабилним конфигурацијама произвољно дуго. Како систем онда прелази из једног таквог стања у друго?

Пошто се термална енергија $k_b T$ може занемарити, промена може настати само спољашном побудом система, нпр. вибрацијом. У овој ситуацији Мехта и Едвардс (Mehta and Edwards, 1989 [9]) су предложили нови формализам који би променио конвенционалну термодинамику у случају гранула. Они занемарују енергију (пошто претпостављамо да честице интерагују међусобно само путем одбојне силе на површини) и замењују Хамилтонијан запреминским функционалом. Ентропија је и даље логаритам броја стања у датој запремини [12], а такође се и остале термодинамичке величине могу дефинисати на аналоган начин као у „обичној“ термодинамици. Међутим, уместо термалне енергије $k_b T$ нова ефективна температура се уводи на другачији начин.

Проучавања грануларних материјала који су подвргнути спољним вибрацијама показују да је релаксација ових система веома спора [13]. Чак и након 100 000 циклуса, материјал може трпети значајну компакфикацију пре него што густина достигне стационарну вредност. Бројни модели су предложени да би објаснили ову појаву и у њима је вибрациона компакфикација углавном третирана као логаритамски спор процес [2]. Разлог томе је геометријска фрустрација која је инхерентна грануларним системима велике густине. Она нужно индукује кооперативност догађаја на нивоу гранула. Другим речима, један објект (гранула) ће попунити део слободне запремине густог система само уколико дође до кооперативне прерасподеле околних гранула.

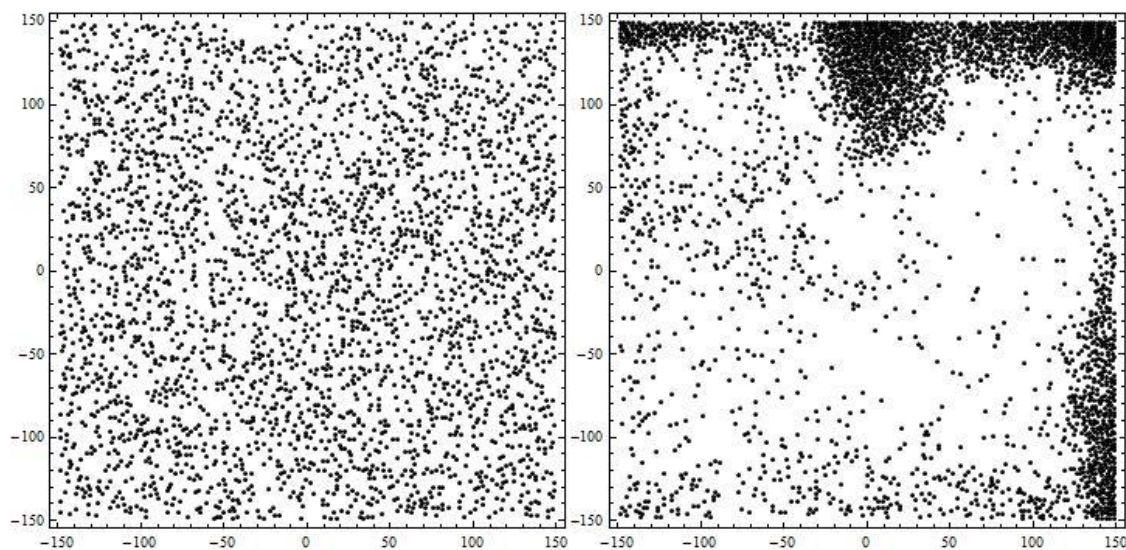
Познато је да динамику течности описујемо Навије – Стоксовим једначинама. У случају грануларних течности оне не важе. Као што смо у уводу рекли ово је последица непостојања разлике у скалама која нам у случају обичних течности омогућава усредњавање. Такође, спори грануларни протоци нису ергодички, а код брзих се јавља проблем кластеровања, ако су судари веома дисипативни, као што ћемо и видети у овом раду.

Случај разређених грануларних материјала који су у динамичком стању веома брзог тока нас ће, у овом раду, највише занимати. Довољно разређени грануларни системи који су добро флуидизирани обично се називају **грануларним гасовима**.

2.2 Основне особине грануларних гасова

За разлику од „обичних“ гасова, грануларни гасови веома брзо губе енергију у међусобним сударима тако да је потребно стално допремати додатну енергију путем спољне побуде да би очували гасовито стање. Овај губитак енергије је последица нееластичности сила међусобне интеракције. Зато, на пример, теорија идеалних гасова, овде није употребљива. Важно је имати у виду да су све појаве динамичке природе тако да је у основи описивања ових система Њутнова динамика. Међутим поред великог броја честица додатни проблем у решавању система једначина ствара и недовољно испитана зависност нееластичности честица и силе трења од релативне брзине и трајања судара [2]. Третман не-чеоних судара као и оних који се дешавају при малим брзинама задају нарочите потешкоће јер тада имамо прелаз са статичког трења на динамичко. Постојање оваквог прелаза показује да су модели са константним коефицијентом реституције само упрошћења реалних процеса.

Осим тога, за разлику од стандардних гасова који временом теже да се „хомогенизују“ и на тај начин смање ентропију, код грануларних се могу јавити различити специфични феномени. То су пре свега „**груписање**“ (кластеровање, clustering eng.) и **смицање** (sheering eng.). Оба ова феномена су предмет истраживања овог рада.



Слика 1.4 – пример почетне расподеле честица гаса (лево) и касније појаве кластера (десно)

Теоријски модели који описују понашање грануларних гасова постоје. Основни закон који ћемо користити и позивати се током рада је **Хафов закон** (P. K. Haff, 1983 [6]) и он се односи на **хомогено** хлађење грануларног гаса, односно процес када је гас препуштен самом себи (изоливан). При појави поменутих феномена кластеровања и смицања, као што ћемо видети, долази до одступања од Хафвог закона.

2.3 Слободно хлађење грануларног гаса

Када грануларни гас препустимо себи, услед дисипативног карактера међусобних судара честица, укупна кинетичка енергија система почиње да се смањује. На основу поменуте могућности дефинисања температуре грануларног система можемо рећи да се систем хлади. То је и главна тема разматрања овог рада. Кинетичка енергија система углавном одлази на деформацију честица и њихово загревање, тако да се важење термодинамике овог система не доводи у питање.

У почетном тренутку сматрамо да се систем налази у крајње случајном (у нашем случају униформном) распореду положаја и брзина честица. Прву фазу временске еволуције система назив **хомогеном** или **кинетичком**. Током ње средња кинетичка енергија система опада, али густина система у сваком делу простора остаје хомогена, а правци брзина потпуно насумични. Она је и најбоље проучена и погодна због свог стохастичког карактера за аналитичко описивање. Зависност промене ново-уведене температуре од времена у хомогеној фази даје поменути Хафов закон који ће бити детаљно описан у једном од наредних поглавља.

Слободни грануларни гас не остаје произвољно дуго у кинетичкој фази. Разлог томе су флукуације које су увек присутне у систему. Оне доводе до појаве нехомогености у пољу брзина и густина честица и тада наступају фазе смицања, односно кластеровања. Ове нестабилности ће такође бити предмет изучавања у овом раду.

За потребу проучавања и приказивања различитих фаза временске еволуције система служили смо се нумеричким симулацијама. За симулацију грануларних система постоје неколико приступа. **Молекуларно динамички** приступ има вероватно највећу улогу међу њима. То је временски зависно, нумеричко решавање Њутнових једначина кретања за све честице од којих се систем састоји. Под одређеним условима, ова метода се може поједноставити ако претпоставимо да у било ком тренутку времена било која честица може бити у контакту са само још једном честицом. То је такозвана апроксимација крутих сфера. Ова апроксимација је оправдана ако је време између судара много веће него време трајања судара. У овом случају **event-driven** молекуларно динамичка метода је нарочито ефикасна. Симулације коришћене у овом раду прате овај алгоритам, који омогућава решавање једначина кретања много већег система честица него нпр. прва метода.

У свим симулацијама смо претпоставили да су све честице сферног (односно у две димензије кружног) облика, једнаких величина и, зависно од услова симулације, се налазе у запремини коначних димензија или пак у области са периодичним граничним условима. Основни параметри које смо мењали су: коефицијент реституције ϵ , густина система честица и број честица.

3 Динамика судара и *event-driven* алгоритам

За разлику од основне *молекуларно динамичке* методе, која се још назива и *метода заснована на сили* (*force-based eng.*) за потребе проучавања грануларних гасова користи се *event-driven* метода [4]. Зашто је то тако?

Прво, системи грануларних гасова су разређени и у њима можемо претпоставити да је средње време између узастопних судара много веће од трајања судара. У том случају већину времена честице проводе крећући се по балистичким путањама које се прекидају само ако честица доживи судар. На тај начин појединачне сударе две честице можемо сматрати тренутним и решавати их одвојено. Ово знатно убрзава молекуларно динамички метод.

Друго, постоје случајеви код којих није позната зависност силе међусобне интеракције као функције релативних положаја $|\vec{r}_1 - \vec{r}_2|$, брзина $|\vec{v}_1 - \vec{v}_2|$, и оријентације честица. Да бисмо знали ту зависност потребно је познавати микроскопски опис честица, што може бити веома компликовано нарочито у случајевима неравномерних облика и нелинеарних материјала. Међутим, могуће је експериментално проучити те сударе и добити зависност која је и потребна за *event-driven* методу, а то је зависност брзина после судара у функцији брзина пре судара.

Основна претпоставка при примени *event-driven* алгоритма је да у сваком тренутку времена постоји највише један судар инфинитезимално малог трајања, у целом систему. Ови судари, као што смо рекли, мењају брзине честица на начин који ћемо детаљно описати (јед. 1.15) и он зависи од коефицијента реституције. Са претпоставком о инфинитезималом трајању и познавајући коефицијент реституције, ове једначине у потпуности описују судар. На тај начин прорачуни путања засновани на решавању Њутнових једначина постају непотребни. Пошто се честице између два судара крећу по правим линијама, оне се могу израчунати у једном кораку. То значајно смањује комплексност, а тиме и време потребно за извршење симулације. У *event-driven* алгоритму динамика система се одређује низом дискретних догађаја.

Осим за разређене системе грануларних гасова, *event-driven* методу можемо применити и код гушћих система. Међутим то није увек оправдано. На пример, у случају челичних сфера на основу *event-driven* алгоритма добијамо да оне не могу проводити струју пошто су у сваком тренутку највише две сфере у контакту [3]. Закључујемо да се оправданост коришћења *event-driven* алгоритма у случају гушћих система мора проверавати од случаја до случаја.

3.1 Судар честица

Релативна брзина честица i и j које учествују у судару у тачки контакта, \vec{g}_{ij} , је одређена транслационом и ротационом брзином честица:

$$\vec{g}_{ij} = (\vec{v}_i - \vec{\omega}_i \times R\vec{e}_{ij}^n) - (\vec{v}_j - \vec{\omega}_j \times R\vec{e}_{ij}^n) = \vec{v}_{ij} - R(\vec{\omega}_i - \vec{\omega}_j) \times \vec{e}_{ij}^n \quad (1.1)$$

где је $\vec{v}_{ij} = \vec{v}_i - \vec{v}_j$, R је полупречник честице, а \vec{e}_{ij}^n јединични вектор који је усмерен од центра честице j ка честици i .

Нормална и тангенцијална компонента релативне брзине су дате следећим изразима:

$$\vec{g}_{ij}^n = (\vec{g}_{ij} \cdot \vec{e}_{ij}^n)\vec{e}_{ij}^n \quad \text{и} \quad \vec{g}_{ij}^t = -\vec{e}_{ij}^n(\vec{e}_{ij}^n \times \vec{g}_{ij}) \quad (1.2)$$

Коефицијент реституције у нормалном и тангенцијалном правцу је дат са:

$$\begin{aligned} (\vec{g}_{ij}^n)' &= -\varepsilon^n \vec{g}_{ij}^n, & \text{за } 0 \leq \varepsilon^n \leq 1 \\ (\vec{g}_{ij}^t)' &= \varepsilon^t \vec{g}_{ij}^t, & \text{за } -1 \leq \varepsilon^t \leq 1, \end{aligned} \quad (1.3)$$

где „примоване“ ознаке означавају величине после судара.

Речено је да брзине након судара можемо написати као функције брзина пре судара и коефицијента реституције. Назваћемо овај скуп функција *закон судара*. Да бисмо га извели потребно је поставити тачку контакта честица за центар ротације. Узимајући да је површина контакта инфинитезимално мала то се може урадити, и тада ни једна сила која делује на честице не изазива момент силе. Последица тога је да не постоји трансфер момента импулса међу честицама. Момент импулса у односу на ову тачку се одржава за сваку честицу понаособ. Дакле, имамо три векторске величине које се одржавају, једна је укупан импулс, а друге две су појединачни моменти импулса:

$$\vec{v}_i' + \vec{v}_j' = \vec{v}_i + \vec{v}_j \quad (1.4)$$

$$mR\vec{e}_{ij}^n \times \vec{v}_i' + J\vec{\omega}_i' = mR\vec{e}_{ij}^n \times \vec{v}_i + J\vec{\omega}_i \quad (1.5)$$

$$mR\vec{e}_{ij}^n \times \vec{v}_j' - J\vec{\omega}_j' = mR\vec{e}_{ij}^n \times \vec{v}_j - J\vec{\omega}_j \quad (1.6)$$

Решавајући (1.5) и (1.6) по $\vec{\omega}_i'$ и $\vec{\omega}_j'$, добијамо:

$$\vec{\omega}_i' = \vec{\omega}_i - \frac{mR}{J}\vec{e}_{ij}^n \times (\vec{v}_i' - \vec{v}_i), \quad \vec{\omega}_j' = \vec{\omega}_j + \frac{mR}{J}\vec{e}_{ij}^n \times (\vec{v}_j' - \vec{v}_j), \quad (1.7)$$

И користећи израз за \vec{g}_{ij}' (који се добија када у јед. 1.1 ставимо примоване ознаке) налазимо:

$$\begin{aligned} \vec{g}_{ij}' &= \vec{v}_{ij}' - R(\vec{\omega}_i' + \vec{\omega}_j') \times \vec{e}_{ij}^n \\ &= \vec{v}_{ij}' - R(\vec{\omega}_i + \vec{\omega}_j) \times \vec{e}_{ij}^n + \frac{mR^2}{J}\vec{e}_{ij}^n \times [\vec{e}_{ij}^n \times (\vec{v}_{ij} - \vec{v}_{ij}')] \end{aligned} \quad (1.8)$$

Одузимајући (1.8) од (1.1) добијамо:

$$\vec{g}'_{ij} - \vec{g}_{ij} = \vec{v}'_{ij} - \vec{v}_{ij} - \frac{1}{\bar{J}} \vec{e}_{ij}^n \times [\vec{e}_{ij}^n \times (\vec{v}'_{ij} - \vec{v}_{ij})], \quad (1.9)$$

где смо са \bar{J} означили редуковани момент инерције $\bar{J} = \frac{J}{mR^2}$.

Користећи овај резултат можемо написати:

$$\vec{e}_{ij}^n \cdot (\vec{g}'_{ij} - \vec{g}_{ij}) = \vec{e}_{ij}^n \cdot (\vec{v}'_{ij} - \vec{v}_{ij}) \quad (1.10)$$

због тога што је трећи члан у једначини (1.9) нормалан на \vec{e}_{ij}^n .

Одговарајући векторски производ даје:

$$\begin{aligned} \vec{e}_{ij}^n \times (\vec{g}'_{ij} - \vec{g}_{ij}) &= \vec{e}_{ij}^n \times \left[\vec{v}'_{ij} - \vec{v}_{ij} - \frac{1}{\bar{J}} \{ \vec{e}_{ij}^n \cdot [\vec{e}_{ij}^n \cdot (\vec{v}'_{ij} - \vec{v}_{ij})] - (\vec{v}'_{ij} - \vec{v}_{ij}) \} \right] \\ &= \frac{\bar{J}+1}{\bar{J}} \vec{e}_{ij}^n \times (\vec{v}'_{ij} - \vec{v}_{ij}) - \frac{1}{\bar{J}} \vec{e}_{ij}^n \times \{ \vec{e}_{ij}^n \cdot [\vec{e}_{ij}^n \cdot (\vec{v}'_{ij} - \vec{v}_{ij})] \} \\ &= \frac{\bar{J}+1}{\bar{J}} \vec{e}_{ij}^n \times (\vec{v}'_{ij} - \vec{v}_{ij}) \end{aligned} \quad (1.11)$$

При том је употребљена репрезентација произвољног вектора \vec{x} :

$$\vec{x} = \vec{e} \cdot (\vec{e} \cdot \vec{x}) + \vec{e} \times (\vec{e} \times \vec{x}) \quad (1.12)$$

Пошто је $\vec{e}_{ij}^n \times \vec{e}_{ij}^n = 0$, други члан у другом реду једначине (1.11) је 0.

Користећи претходне формуле за $\vec{v}'_{ij} - \vec{v}_{ij}$ можемо писати:

$$\begin{aligned} \vec{v}'_{ij} - \vec{v}_{ij} &= \vec{e}_{ij}^n \cdot [\vec{e}_{ij}^n \cdot (\vec{g}'_{ij} - \vec{g}_{ij})] + \frac{\bar{J}}{1+\bar{J}} \vec{e}_{ij}^n \times [\vec{e}_{ij}^n \times (\vec{g}'_{ij} - \vec{g}_{ij})] \\ &= (\vec{g}'_{ij})' - \vec{g}_{ij} + \frac{\bar{J}}{1+\bar{J}} [(\vec{g}'_{ij})' - \vec{g}_{ij}^t] \\ &= -(1 + \varepsilon^n) \vec{g}_{ij} + \frac{\bar{J}}{1+\bar{J}} (\varepsilon^t - 1) \vec{g}_{ij}^t \end{aligned} \quad (1.13)$$

Ако искористимо дефинициону формулу за $\vec{v}'_{ij} = \vec{v}'_i - \vec{v}'_j$ и једначину (1.4) следи:

$$\begin{aligned} 2\vec{v}'_i &= 2\vec{v}_i + \vec{v}'_{ij} - \vec{v}_{ij} \\ 2\vec{v}'_j &= 2\vec{v}_j + \vec{v}_{ij} - \vec{v}'_{ij} \end{aligned} \quad (1.14)$$

Након што смо већ написали изразе за $\vec{\omega}'_{ij}$ можемо користећи формуле (1.13), (1.14) и израз $\vec{e}_{ij}^n \times \vec{g}'_{ij} = 0$ извести коначне формуле:

$$\vec{v}'_i = \vec{v}_i - \frac{1 + \varepsilon^n}{2} \vec{g}_{ij}^n + \frac{\tilde{J}(\varepsilon^t - 1)}{2(\tilde{J} + 1)} \vec{g}_{ij}^t$$

$$\vec{v}'_j = \vec{v}_j + \frac{1 + \varepsilon^n}{2} \vec{g}_{ij}^n - \frac{\tilde{J}(\varepsilon^t - 1)}{2(\tilde{J} + 1)} \vec{g}_{ij}^t$$

$$\vec{\omega}'_i = \vec{\omega}_i - \frac{\varepsilon^t - 1}{2R(\tilde{J} + 1)} (\vec{e}_{ij}^n \times \vec{g}_{ij}^t)$$

$$\vec{\omega}'_j = \vec{\omega}_j - \frac{\varepsilon^t - 1}{2R(\tilde{J} + 1)} (\vec{e}_{ij}^n \times \vec{g}_{ij}^t)$$

(1.15)

Да ли је овај систем јединствен? Да би био, на почетку је било потребно да имамо 12 скаларних једначина.

Три једначине су добијене из закона одржања импулса. Из закона одржања момента импулса још шест, јер смо имали претпоставку о крутим сферама, која је имплицирала инфинитезимално малу додирну површину, и омогућила да напишемо законе о одржању момента импулса за сваку честицу понаособ. Ти недостајуће једначине су добијене из разматрања о енергији.

Закон одржања механичке енергије генерално не важи, већ смо увели у разматрање и коефицијент реституције. Увели смо посебно коефицијенте ε^n и ε^t који се односе на губитке при нормалном кретању и тангенцијалном. Међутим да ли познајући ова два коефицијента добијамо три једначине? Да, јер ако уведемо координатни систем где је z -оса дуж правца који повезује центре куглица, онда се ε^n односи на тај правац. Даље, можемо рећи да је x -оса нормална на тај правац, и усмерена дуж правца тангенцијалне брзине, и остаје нам да је брзина дуж y -осе 0 . Таква је и после судара. И то нам даје додатни скуп од три једначине. Тако да заиста имамо 12 независних услова који нам гарантују јединственост горњег система једначина.

И наведимо још апроксимације које смо користили. Прва је да је површина контакта сматра тачком. Могућност је била и да је сматрамо коначном површином, што би одговарало случају „меких“ сфера, али такав случај још увек није у потпуности решен [4]. А друга је да су кретања, односно губици енергије дуж два нормална правца независни, што такође није случај у реалности, већ су ε^n и ε^t међусобно зависни.

3.2 Нацрт *event-driven* алгоритма

Дакле, имајући систем једначина (1.15) можемо скицирати наш алгоритам:

1. Свакој честици доделимо скуп координата положаја \vec{r}_i , брзина \vec{v}_i , и угаоних брзина $\vec{\omega}_i$ у тренутку $t = 0$, тј. иницијализација положаја и брзина ³
2. Одређује се време следећег судара $t^* > 0$, тј. време :

$$t^* = \min(t_{ij} > 0: |\vec{r}_i(t_{ij}) - \vec{r}_j(t_{ij})| = 2R), \quad i, j = 1, 2, 3, \dots, N \quad (1.16)$$

3. Одређује се положај сваке честице у тренутку t^* :

$$\vec{r}_i := \vec{r}_i + (t^* - t)\vec{v}_i, \quad \text{за } i = 1, \dots, N \quad (1.17)$$

(ова зависност се може лако генерализовати и у случају гравитационог или неког другог спољашњег поља)

4. Рачунају се брзине и угаоне брзине након судара честица које су учествовале у судару на основу скупа једначина (1.15)

$$\begin{aligned} \vec{v}_i &:= \vec{v}_i(\vec{v}_i, \vec{v}_j, \vec{\omega}_i, \vec{\omega}_j), \\ \vec{v}_j &:= \vec{v}_j(\vec{v}_i, \vec{v}_j, \vec{\omega}_i, \vec{\omega}_j), \\ \vec{\omega}_i &:= \vec{\omega}_i(\vec{v}_i, \vec{v}_j, \vec{\omega}_i, \vec{\omega}_j), \\ \vec{\omega}_j &:= \vec{\omega}_j(\vec{v}_i, \vec{v}_j, \vec{\omega}_i, \vec{\omega}_j) \end{aligned} \quad (1.18)$$

5. Ажурирати време:

$$t := t^*$$

6. Наставити са тачком 2.

И тиме је алгоритам завршен.

³ у овом раду није узимано у разматрање угаоно кретање

Упоредјујући овај алгоритам са поменутиим *молекулско динамичким* алгоритмом заснованим на *сили* можемо поновити предности *event-driven* алгоритма :

- Није потребно познавати силу интеракције између честица као функцију брзина и угаоних брзина, већ на основу експерименталних података одредити коефицијенте ε^n и ε^t као функције релативне брзине \vec{g}_{ij} , и користити их на начин што бољег фитовања на податке
- Комплексност симулације је одређена бројем судара. Није потребно додатно ангажовање за рачунање положаја и брзина честица између судара. За разлику од алгоритма заснованог на *сили*, *event-driven* систем се не креће по унапред задатим корацима времена, већ од судара до судара. Зато се и назива *event-driven* алгоритам. Можемо још поменути да што је мања густина система ефикасност овог алгоритма постаје већа.

Оправданост алгоритма се заснива на премиси да у једном тренутку можемо имати само један судар, односно да је време судара веома мало. То можемо сматрати делимично испуњеним, јер време судара у реалности ипак има одређено трајање. За еластичне сударе можемо писати [4]:

$$t_{coll}^{el} \propto t^{-1/5},$$

а нееластични трају чак и дуже.

4 Реализација кода на основу *event-driven* алгоритма

Реализација кода на основу *event-driven* алгоритма је захтеван подухват. У овом раду је коришћен код који је највећим делом презентован у књизи „*Computational Granular Dynamics*“ (Thorsten Pöschel, Thomas Schwager [4]). На почетку ћемо изнети апроксимације које су коришћене у изради кода.

Прво, грануле су кружног облика и међусобно једнаке. Следећа апроксимација је да се ротациони степени слободе могу занемарити (нема трења, па ни преноса момента импулса међу честицама). На основу тога, за нас је значајан само нормалан коефицијент реституције ε^n . Овакав опис система се користи често и у кинетичкој теорији грануларних гасова [14].

Ми смо, такође, у овом раду већином користили чврсте зидове, односно ограничену област квадратног облика, дужине $2*Lbox$, тј., наша област је имала координате темена: $(-Lbox, -Lbox)$, $(Lbox, -Lbox)$, $(-Lbox, Lbox)$, $(Lbox, Lbox)$. Међутим, искористили смо и могућност увођења периодичних граничних услова, тако да су и они примењени у једном делу рада. Напоменимо и да се код може проширити на честице различитих полупречника R_i , као и на случај три димензије.

Сада ћемо изложити код израђен у програмском језику C++, преко функција које представљају имплементацију претходно поменутих физичких закона и *event-driven* алгоритма.

4.1 Кретање честица по инерцији између судара

Честице се у одсуству спољњег поља крећу по правим линијама у времену $tprop$ у којем нема судара. Током тог времена растојање које пређу је просто: $\vec{r}_i = \vec{v}_i t_{prop}$. Ово је имплементирано у функцију *propagation()*.

```
void propagation(double tprop)
{
    for(int i=0; i<N; i++){
        x[i]+=tprop*vx[i];
        y[i]+=tprop*vy[i];
    }
}
```

4.2 Сударање две честице

Закон судара (1.15) нам омогућава да направимо функцију која би рачунала брзине после судара на основу података о брзинама пре. Он се упрошћава нашим апроксимацијама, па за израду кода користимо једначине:

$$\begin{aligned}\vec{v}_i &= \vec{v}_i - \frac{1+\epsilon}{2} [(\vec{v}_i - \vec{v}_j) \cdot \vec{e}_{ij}^n] \vec{e}_{ij}^n \\ \vec{v}_j &= \vec{v}_j + \frac{1+\epsilon}{2} [(\vec{v}_i - \vec{v}_j) \cdot \vec{e}_{ij}^n] \vec{e}_{ij}^n\end{aligned}\quad (4.1)$$

Судар две честице рачуна функција *collision()*.

```
void collision(const pair<int,int> & ij)
{
    int i=ij.first;
    int j=ij.second;
    double dx = x[i]-x[j];
    double dy = y[i]-y[j];
    double dist=sqrt(dx*dx+dy*dy);
    double ndx=dx/dist;
    double ndy=dy/dist;
    double h=(1+eps) * ((vx[i]-vx[j])*ndx+(vy[i]-vy[j])*ndy)/2;

    vx[i]-=h*ndx;
    vy[i]-=h*ndy;
    vx[j]+=h*ndx;
    vy[j]+=h*ndy;
}
```

Индекси *i* и *j* означавају појединачне честице које учествују у судару, а $vx[i]$, $vy[i]$, $vx[j]$, $vy[j]$ пројекције брзина које модификујемо услед судара. Да бисмо одредили када ће доћи до судара и које две честице ће се следеће сударити, прво одређујемо да ли ће се било које две честице сударити ако наставе да се крећу по правим линијама константним брзинама. Ако ће се сударити једначина

$$|[\vec{r}_i + (t^* - t)\vec{v}_i] - [\vec{r}_j + (t^* - t)\vec{v}_j]| = R_i + R_j, \quad (4.2)$$

има решење $t^* > 0$, где је t реално време. Са ознакама $\vec{r}_{ij} = \vec{r}_i - \vec{r}_j$, $\vec{v}_{ij} = \vec{v}_i - \vec{v}_j$, $R_{ij} = R_i + R_j$ једначина (4.2) постаје:

$$(t^* - t)^2 + 2(t^* - t) \frac{\vec{r}_{ij} \cdot \vec{v}_{ij}}{v_{ij}^2} + \frac{r_{ij}^2 - R_{ij}^2}{v_{ij}^2} = 0 \quad (4.3)$$

Да решење (4.3) постоји потребно је да буде испуњен услов:

$$\left[\frac{\vec{r}_{ij} \cdot \vec{v}_{ij}}{v_{ij}^2} \right]^2 + \frac{R_{ij}^2 - r_{ij}^2}{v_{ij}^2} > 0 \quad (4.4)$$

Потребан услов да би честице могле да се сударе је:

$$\vec{r}_{ij} \cdot \vec{v}_{ij} < 0 \quad (4.5)$$

Када су испуњени ови услови честице се сударају у тренутку t^* .

Време судара t^* рачуна функција *ppcoll()*, ако дође до судара, у супротном бележи вредност *infty*, односно веома велику вредност коју смо раније унели. Вредност *infty* се такође бележи ако је место судара изван наше области. Да бисмо добили листу свих времена судара и одредили најмање, односно онај судар који ће бити следећи користимо функцију *update()*.

```

double ppcoll(int i, int j, double tol)
{
    double dx, dy, dvx, dvy, xci, xcj, yci, ycj, scalar, h, p, q, w, ct;
    double dist2, RR, vv2;

    dx=x[i]-x[j];
    dy=y[i]-y[j];
    dvx=vx[i]-vx[j];
    dvy=vy[i]-vy[j];
    scalar=dx*dvx+dy*dvy;
    if(scalar>0){
        return infty;
    } else{
        dist2 = dx*dx+dy*dy;
        RR = (dist2>=R*R ? R : R-tol);
        vv2 = dvx*dvx+dvy*dvy;
        q = dist2-4*RR*RR;
        w = scalar*scalar - q*vv2;
        if(w<0){
            return infty;
        } else {
            ct=Time+q/(-scalar + sqrt(w));
            xci=x[i]+(ct-Time)*vx[i]; xcj=x[j]+(ct-Time)*vx[j];
            if((xci>Lbox-R)|| (xci<-Lbox+R)|| (xcj>Lbox-R)|| (xcj<-Lbox+R)){
                return infty;
            } else {
                yci=y[i]+(ct-Time)*vy[i]; ycj=y[j]+(ct-Time)*vy[j];
                if((yci>Lbox-R)|| (yci<-Lbox+R)|| (ycj>Lbox-R)|| (ycj<-Lbox+R)){
                    return infty;
                } else {
                    return ct;
                }
            }
        }
    }
}

```

Да би избегли нумеричке грешке потребно је дефинисати и веома мали број *tol* који се користи као трећи аргумент функције *ppcoll()*.

4.3 Судари са зидовима

Ако се претпостави да се грануле еластично сударају са зидовима, можемо писати функцију *wcollision()*:

```
void wcollision(int i)
{
    if( y[i]<0){
        if((y[i] > x[i]) || (y[i] > -x[i])){
            vx[i]=-vx[i];
        } else{
            vy[i]=-vy[i];
        }
    } else {
        if((y[i] < -x[i]) || (y[i] < x[i])){
            vx[i]=-vx[i];
        } else{
            vy[i]=-vy[i];
        }
    }
}
```

Такође, као и у случају судара честица, морамо израчунати време судара са зидом. То ћемо израчунати као време потребно да се било која честица судари са зидом, и онда наћи минимум тих времена. То рачуна функција *pwcoll()*.

```
double pwcoll(int i, double tol)
{
    double tx, ty;
    double RR = R;

    if((Lbox+x[i]>R) || (Lbox-x[i]<R) || (Lbox+y[i]>R) || (Lbox-y[i]<R)) {
        RR=R-tol;
    }
    if(vx[i]==0){
        tx=infty;
    } else {
        if(vx[i] >0) {
            tx=(Lbox-x[i]-RR)/vx[i];
        } else {
            tx=(-Lbox-x[i]+RR)/vx[i];
        }
    }
    if(vy[i]==0){
        ty=infty;
    } else {
        if(vy[i] >0){
            ty=(Lbox-y[i]-RR)/vy[i];
        } else {
            ty=(-Lbox-y[i]+RR)/vy[i];
        }
    }
    return Time+min(tx,ty);
}
```

4.4 Периодични гранични услови

Као што смо рекли део симулација је извршен и са периодичним граничним условима. У суштини, потребно је само уместо функције за чврсте зидове убацити измењену функцију *wcollision()*:

```
void wcollision(int i)
{
    bool check = false;
    if( x[i]+Lbox-null<0 || x[i]-Lbox+null>0){
        x[i]+=2*sign(-x[i])*Lbox;
        check = true;
    }

    if( y[i]+Lbox-null<0 || y[i]-Lbox+null>0){
        y[i]+=2*sign(-y[i])*Lbox;
        check = true;
    }
}
```

А, функција *sign()* је дата са:

```
double sign(double x){
    if(x>0)
        return 1;
    else if (x<0)
        return -1;
    else return 0;
}
```

4.5 Распоређивање почетних положаја и брзина

У нашем коду почетне брзине и положаје смо добили коришћењем функције *ranf()*.

```
double ranf(double x){ return (2*double(rand())/(1+double(RAND_MAX))-1)*x; }
```

Пошто се водило рачуна о томе да не може бити преклапања честица дата је целокупна функција *init()* која распоређује почетне параметре.

```

void init(double tol)
{
    bool overlap;
    int j;

    x[0]=ranf(Lbox-R-tol);
    y[0]=ranf(Lbox-R-tol);
    vx[0]=ranf(1);
    vy[0]=ranf(1);
    for(int i=1; i<N; i++){
        if(!(i % 100 )) cout << "Init " << i << endl;
        do{
            overlap=false;
            x[i]=ranf(Lbox-R-tol); y[i]=ranf(Lbox-R-tol);
            j=0;
            do{
                overlap=((x[i]-x[j])*(x[i]-x[j])+(y[i]-y[j])*(y[i]-y[j])<
                    4*(R+tol)*(R+tol));
            } while(++j<i) && !overlap);
        } while(overlap);
        vx[i]=ranf(1);
        vy[i]=ranf(1);
    }
    for(int i=0; i<N; i++) update(i);
}

```

Ова, релативно једноставна функција има нумеричку комплексност $O(N^2)$, Па је примењива само на разређене системе чији број честица не прелази 10 000. У нашем раду, ми смо већином радили са системима од 2600 честица.

4.6 Утврђивање распореда судара

Као што смо већ рекли при судару две честице, да бисмо знали који судар ће се одиграти морамо наћи минимум времена претпостављених судара. Међутим, када имамо и чврсте зидове морамо и те сударе узети у обзир, тако да за следећи судар добијамо онај чије је укупно време минимално (рачунајући међусобне сударе и сударе са зидовима). Можемо истаћи и број таквих потенцијалних догађаја. Наиме, ако бисмо просто рачунали број свих судара имали бисмо $\frac{N(N-1)}{2} + N$. Ово је веома велики број имајући у виду да је то број из кога би требало да се добије време за само један следећи судар.

Међутим, постоје и ефикаснији начини да се добије време t^* . Можемо направити листу свих судара, и онда, након следећег судара није потребно мењати читаву листу, већ само оне чланове у којима фигуришу честице (или само једна честица) које су учествовале у судару.

Све ово рачуна функција `update()`:

```
void update(int i)
{
    double ct;

    for(unsigned int ii=0; ii!=clist[i].size(); ii++)
cseq.erase(clist[i][ii]);
    clist[i].clear();
    for(int j=0; j<N; j++){
        if(i!=j){
            ct=ppcoll(i,j,null);
            if(ct < infty){
                cseq[ct]=pair <int, int> (i,j);
                clist[i].push_back(ct);
                clist[j].push_back(ct);
            }
        }
    }
    ct=pwcoll(i,null);
    if(ct < infty){
        cseq[ct]=pair<int, int>(i,-1);
        clist[i].push_back(ct);
    }
}
```

4.7 Главни програм

Сада, пошто имамо све потребне функције, можемо приступити изради главног програма. Користићемо следеће ознаке: ***N*** – број честица, ***nstep*** – укупан број судара који ће се рачунати, ***nps*** – интервал у којем ће се прикупљати један део података, ***nprint*** – такође интервал за прикупљање података, ***noverlap*** – интервал у ком се проверава да ли долази до преклапања, ***pssize*** – величина snapshot-ova, ***R*** – полупречник честица, ***eps*** – коефицијент реституције, ***Lbox*** – број који одређује величину области ($-Lbox, Lbox$), ***infy*** - веома велики број који представља бесконачност, ***null*** – мали број који служи да би се избегле нумеричке грешке.

```

#include <iostream>
#include <fstream>
#include <vector>
#include <map>
#include <cstdlib>
#include <cmath>

using namespace std;

const int N=1000, nstep=5000000, nps=10000, nprint=100, noverlap=1000;
const double R=1, eps=0.95, Lbox=150, pssize=500;
const double infity=1e20, null=1e-10;

vector<double> x(N), y(N), vx(N), vy(N);
vector<double> clist[N];
double Time=0;
map<double, pair<int, int> > cseq;

ofstream fps("event.ps"), fenergy("kinenergy");

void init(double);
void propagation(double);
void collision(const pair<int, int> &);
void wcollision(int);
void update(int);
void psplot(int, ofstream &);
bool checkoverlap();
double ppcoll(int, int, double);
double pwcoll(int, double);
double ranf(double x){ return (2*double(rand())/(1+double(RAND_MAX))-1)*x;}
double kinenergy();

```

У главном програму у свакој итерацији, време t_{next} , и ознаке честица i и j које ће се следеће сударити се рачунају као први члан из листе свих судара $cseq$. Ако је ознака другог члана у пару $\langle i, j \rangle$ једнака -1 , то значи да долази до судара са зидом. Потом се позива функција $propagation()$, која одређује положаје свих честица до тренутка судара. Даље рачунамо брзине након судара честице/а које учествују у судару на основу функције $collision()$, односно $wcollision()$ зависно од типа судара. И потом се позива функција $update()$ која ће обновити листу $cseq$

```

int main()
{
    double tnext, tn=0;
    pair<int,int> ijnext;

    init(null);

    for(int it=0; it<nstep; it++){
        if(it % noverlap==0) checkoverlap();
        if(it % nprint ==0) fenergy << it<<" "<<kinenergy() << endl;
        tnext=cseq.begin()->first;
        if(!(it%nprint)) cout << "IT: " << it << " time= " << Time << endl;
        ijnext=cseq.begin()->second;
        if(ijnext.second==-1){
            tn=pwcoll(ijnext.first,null);
            propagation(tn-Time);
            Time=tnext;
            wcollision(ijnext.first);
            update(ijnext.first);
        } else{
            tn=ppcoll(ijnext.first,ijnext.second,null);
            propagation(tn-Time);
            Time=tnext;
            collision(ijnext);
            update(ijnext.first);
            update(ijnext.second);
        }
        if(it%nps==0) psplot(it/nps,fps);
    }
}

```

Можемо приметити да се у главном делу програма поново рачуна време судара. Ово је урађено да би се избегла нумеричка грешка до које може доћи као последица нагомилавања малих одступања услед великог броја судара.

4.8 Прикупљање и обрада података

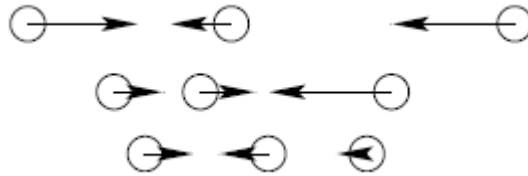
Подаци су прикупљани у оквиру текстуалног фајла у виду табеле у којој су били подаци о положају сваке честице након одређеног броја судара, онда о вредности компонената брзине, кинетичкој енергији, и броју судара које је претрпела свака честица.

Потом су ови подаци обрађивани помоћу програма *Wolfram Mathematica* и *MATLAB* и на основу тих програма су добијени визуелни прикази добијених резултата.

5 Појава нееластичног колапса у нумеричким симулацијама

При конкретној реализацији кода, у одређеним случајевима, долази до појаве која се назива *нееластичан колапс*. То је појава да се група честица почне сударати бесконачно често у коначном интервалу времена [5]. Ово је разлог због којег *event-driven* алгоритам једноставно престане да даље еволуира систем.

Можемо размотрити услов настанка нееластичног колапса у једној димензији. Нека имамо најростији случај три честице које се крећу дуж једне праве. Ако је коефицијент реституције довољно мали, можемо имати случај као на слици и тада се време између судара смањује и након бесконачно много судара (идеализација) систем наставља да се креће као целина заједничком брзином.



Слика 5.1 – приказ ситуације која доводи до нееластичног колапса три честице

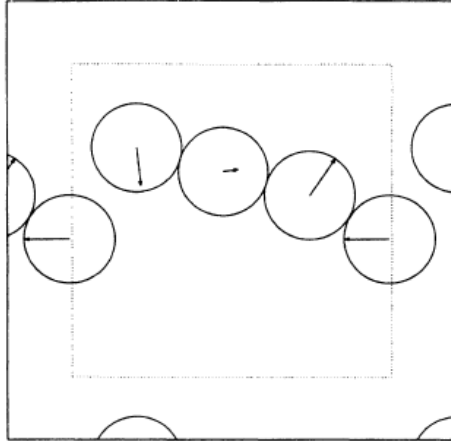
Услов за нееластичан колапс у једној димензији за три честице је [5]:

$$0 < \varepsilon < 7 - 4\sqrt{3} \cong 0,07$$

И као што видимо, потребан је веома нееластичан систем да би се ово догодило. Са четири честице је могуће и са већим ε :

$$0 < \varepsilon < 3 - 2\sqrt{2} \cong 0,17$$

и тако даље, што је већи број честица могуће је да дође до нееластичног колапса и при врло великим коефицијентом реституције, односно при доста еластичним сударима. У две димензије је такође могуће да дође до нееластичног колапса (слика 5.2). Ово нам је и стварало проблем приликом конкретне реализације и еволуције различитих система о чему ћемо касније причати. Може доћи и до *непотпуног* нееластичног колапса када систем делимично, у одређеном интервалу времена, бива „закочен“.

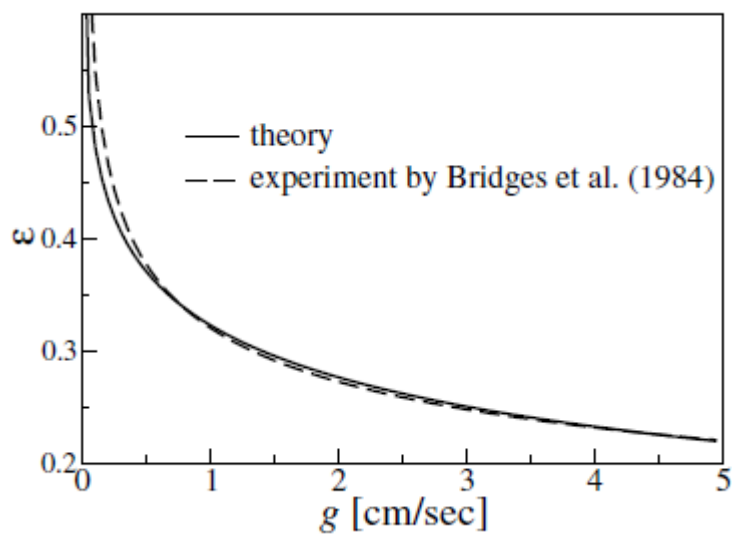


Слика 5.2 – 4 диска у тренутку нееластичног колапса, у систему са периодичним граничним условима

5.1 Коэффициент реституције

Истраживања показују да коэффициент реституције није константан, већ у мањој или већој мери зависи од релативне брзине честица. Постоје бројни радови и истраживања на ову тему [4], [6], а ми ћемо се овде укратко осврнути на неке од њих.

Једно од истраживања је вршено над сферама од леда, на веома ниској температури, и добијена је зависност нормалног коефицијента реституције од релативне брзине. Ова зависност је приказана на слици 5.3.

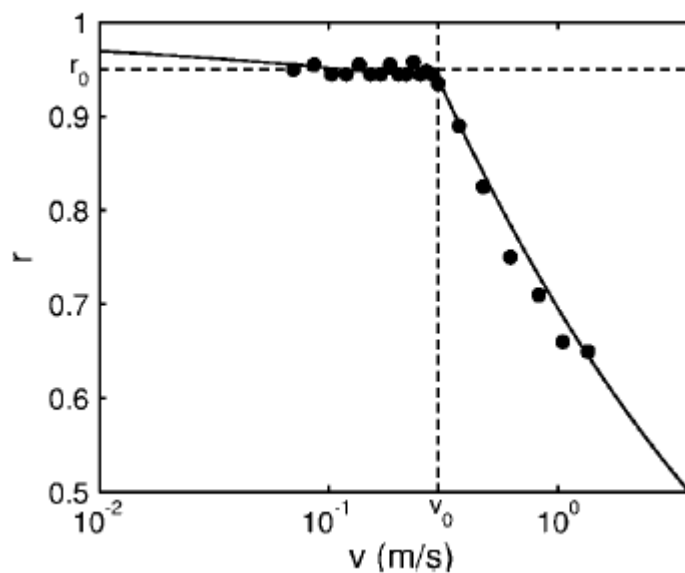


Слика 5.3 – нормални коэффициент реституције у функцији од нормалне брзине судара. Испрекидана линија: експериментални подаци, пуна линија: теоријски резултати

Поред тога у раду [7] вршено је истраживање над челичним куглицама и добијена је емпиријска формула:

$$r(v) = \begin{cases} 1 - (1 - r_0) \left(\frac{v}{v_0}\right)^{\frac{1}{5}}, & v \leq v_0 \\ r_0 \left(\frac{v}{v_0}\right)^{-1/4}, & v \geq v_0 \end{cases} \quad (5.1)$$

која се слаже са експериментима (слика 5.4).



Слика 5.4 – коефицијент реституције r у зависности од брзине судара v . Пуна линија је добијена са параметрима $v_0 = 0.3$ и $r_0 = 0.95$. Пуни кружићи представљају експерименталне податке

У овом раду ћемо се углавном бавити константним коефицијентом реституције, а до промене може доћи само да би се избегао нееластични колапс.

5.2 Методе решавања нееластичног колапса

Постоје различите методе за избегавање нееластичног колапса и сада ћемо навести неке, које смо такође користили и при реализацији кодова:

- Први начин се односи на коефицијент реституције. У следећем поглављу ћемо рећи нешто више о зависности ε од релативне брзине, а сада можемо рећи да постоји начин да се нееластични колапс избегне тиме што ће се, када релативна брзина судара падне испод неке вредности, ставити да коефицијент реституције узима неку нову вредност блиску јединици. Ово је више налик ономе што се догађа у експериментима јер као што смо видели $\varepsilon_{g \rightarrow 0} = 1$.
- Други начин је такозвани ТС – модел [4], када претпостављамо да се судар одвија унутар временског интервала t_c , и ако се следећи судар јави за време мање од t_c , онда стављамо $\varepsilon = 1$, тј. судар постаје еластичан.
- Трећи начин превазилажења нееластичног колапса је да након судара векторе брзина честица заротирамо за произвољан мали угао из неког интервала (нпр. $0^\circ - 5^\circ$). На тај начин се део нормалне брзина трансформише у тангенцијалну и то прекида низ судара.

У овом раду је већина резултата добијена на основу методе која комбинује први и трећи случај. Конкретан код се добија тако што се функција *collision()* делимично измени.

6 Кинетичка теорија грануларних гасова и Хафов закон

На основу кинетичке теорије грануларних гасова, хомогено стање гаса (кинетичку фазу) можемо описати Хафовим законом [6]:

$$T_h(t) = T_h(0) \left[1 + \frac{t}{t_0} \right]^{-2}. \quad (6.1)$$

Ми смо током рада са симулацијама за сваки појединачни систем, на основу посебног програма израђеног у *MATLAB*-у, цртали криву коју предвиђа овај закон и резултате симулација поредили са њом.

Сада ћемо детаљније рећи шта представља сваки члан у овој формули. Прво, уводимо кинетичку температуру $T_h(t)$ на следећи начин:

$$\frac{d}{2} n_H k_b T_h(t) = \int d\vec{v} \frac{1}{2} m v^2 f_h(\vec{v}, t), \quad (6.2)$$

односно,

$$\frac{d}{2} k_b T_h(t) = \frac{1}{2} m \frac{1}{n_H} \int d\vec{v} * v^2 f_h(\vec{v}, t) = \frac{1}{2} m \langle v^2 \rangle_H, \quad (6.3)$$

где су d – димензија система, \vec{v} – брзина честица, k_b – Болцманова константа, m – маса појединачне честице, а $\langle v^2 \rangle_H$ – брзина усредњена по једночестичној функцији расподеле.

Ако претпоставимо да једночестична функција расподеле грануларног система у хомогеном стању хлађења (HCS) има облик [7]:

$$f_h(\vec{v}, t) = \frac{n_H}{v_0^d(t)} \Phi \left[\frac{\vec{v}}{v_0(t)} \right] = \frac{n}{v_0^d(t)} \Phi(v^*), \quad (6.4)$$

где је $\vec{v}^* = \frac{\vec{v}}{v_0(t)}$, а $v_0(t) = \left[\frac{2k_b T_h(t)}{m} \right]^{1/2}$.

$\Phi(v^*)$ се може одредити апроксимативно, из Болцманове (Boltzmann) кинетичке једначине развојем по Сониним (Sonine) полиномима.

Тада хидродинамичка једначина за унутрашњу енергију грануларног система у HCS има облик [6]:

$$\frac{\partial}{\partial t} T_h(t) + T_h(t) \xi_H(t) = 0, \quad (6.5)$$

где су

$$\xi_H(t) = \frac{\pi^{\frac{d-1}{2}} \sigma^{d-1} (1-\varepsilon^2)}{2d\Gamma\left(\frac{d+3}{2}\right)} n_H v_0(t) I_0, \quad (6.6)$$

$$I_0 = \kappa_d(n_H, \sigma) \int d\vec{v}_1^* \int d\vec{v}_2^* (g^*)^3 \Phi(\vec{v}_1^*) \Phi(\vec{v}_2^*), \quad (6.7)$$

уз ознаку $g^* = |\vec{v}_1^* - \vec{v}_2^*|$ и κ_d за парну корелациону функцију.

Парна корелациона функција (*pair correlation function at contact*) је последица чињенице да честице (грануле) заузимају неку **коначну** запремину. Грубо говорећи функција $\kappa_d(n_H, \sigma)$ представља меру одступања густине система од униформне расподеле у некој тачки због присуства једне честице фиксиране у координатном почетку. Суштински, у кинетичкој теорији функција $\kappa_d(n_H, \sigma)$ се јавља као нека врста геометријског фактора који коригује колизиону фреквенцију честица система.

Решење једначине (6.5) је управо Хафов закон (6.1), уз ознаку $t_0 = \frac{2}{\xi_H(0)}$.

$\xi_H(0)$ можемо представити као [8]:

$$\xi_H(0) = (1 - \varepsilon^2) \frac{\sqrt{2}\pi^{\frac{d-1}{2}}}{d\Gamma\left(\frac{d}{2}\right)} n_H \sigma^{d-1} v_0(0) \kappa_d(n_H, \sigma) \left[1 + \frac{3c^*(\varepsilon)}{32}\right], \quad (6.8)$$

$$\text{где је } c^*(\varepsilon) = \frac{32(1-\varepsilon)(1-2\varepsilon^2)}{9+24d+(8d-41)\varepsilon+30\varepsilon^2(1-\varepsilon)}. \quad (6.9)$$

6.1 Нова временска скала

За потребе поређења резултата различитих система показује се погодно да уместо реалног времена t , уведемо *нову временску скалу* чија ће величина τ да буде број судара по честици. Ово је такође погодно јер су резултати *event-driven* алгоритма природно дати управо у тој скали.

Дакле, уводимо време као

$$\tau = \frac{N_c(0,t)}{N} = \frac{1}{2} \int_0^t dt' v_0(t'), \quad (6.8)$$

где је $v_0(t')$ колизиона фреквенција дата следећом једначином [9 ,10]:

$$v_0(t') = \frac{2\pi^{\frac{d-1}{2}}}{\Gamma\left(\frac{d}{2}\right)} \kappa_d(\rho) n_H \sigma^{d-1} \left[\frac{k_b T_H(t)}{m}\right]^{\frac{1}{2}} = \frac{\sqrt{2}\pi^{\frac{d-1}{2}}}{\Gamma\left(\frac{d}{2}\right)} \kappa_d(\rho) n_H \sigma^{d-1} v_0(t) \quad (6.9)$$

Даље пишемо:

$$\tau = \frac{1}{2\gamma} \ln \left(1 + \frac{\xi_H(0)}{2} t \right) = \frac{1}{2\gamma} \ln \left(1 + \frac{t}{t_0} \right), \quad (6.10)$$

где је $\gamma = \gamma_0(1 + \Delta)$, $\gamma_0 = \frac{1-\alpha^2}{2d}$, $\Delta = \frac{3c^*(\alpha)}{32}$

Лако се показује да важи:

$$\exp(2\gamma\tau) = 1 + \frac{t}{t_0}, \quad (6.11)$$

па коначно можемо писати Хафов закон у новом облику:

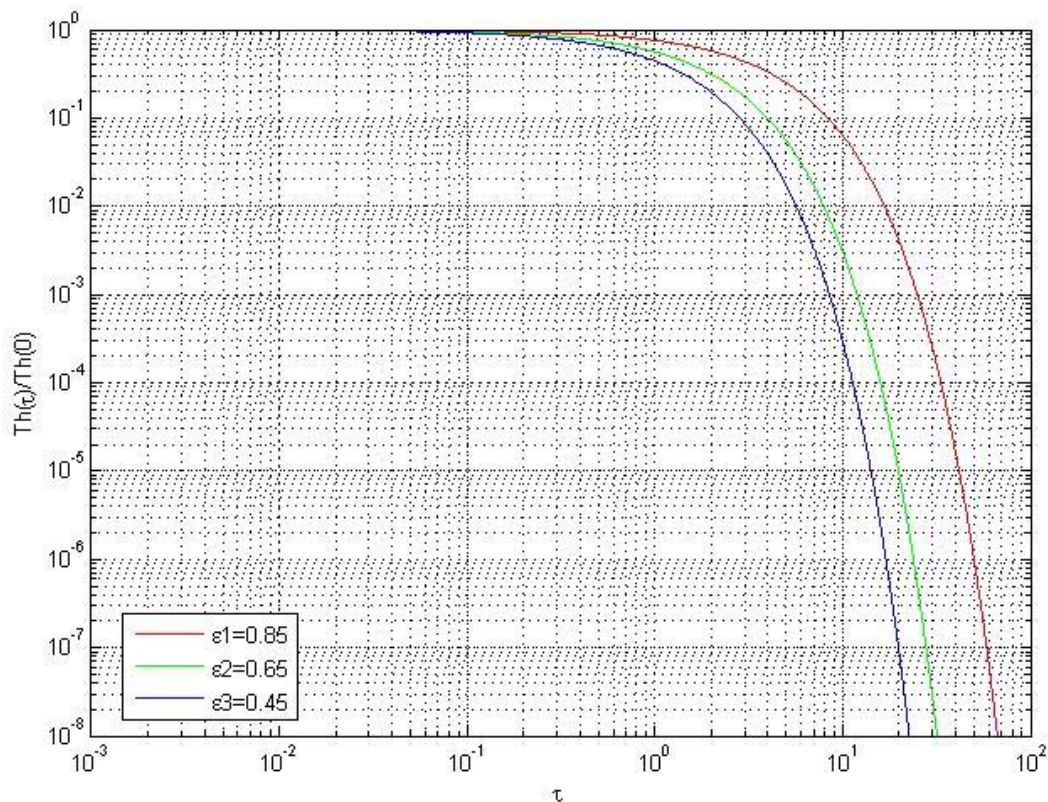
$$T_H(\tau) = T_H(0) * \exp(2\gamma\tau) \quad (6.12)$$

Други део - Презентовање и анализа резултата

7 Поређење различитих метода избегавања нееластичног колапса у нумеричким симулацијама

Као што смо напоменули у поглављу 5.2, у изради овог рада коришћено је више метода избегавања нееластичног колапса, и они су дали донекле различите резултате који ће сада бити презентовани. Ове разлике се односе на касну фазу еволуције система, када су средње енергије честица веома мале. Сви резултати симулација су упоређивани са теоријски добијеним кривама на основу Хафовог закона (6.12) за дати скуп параметара.

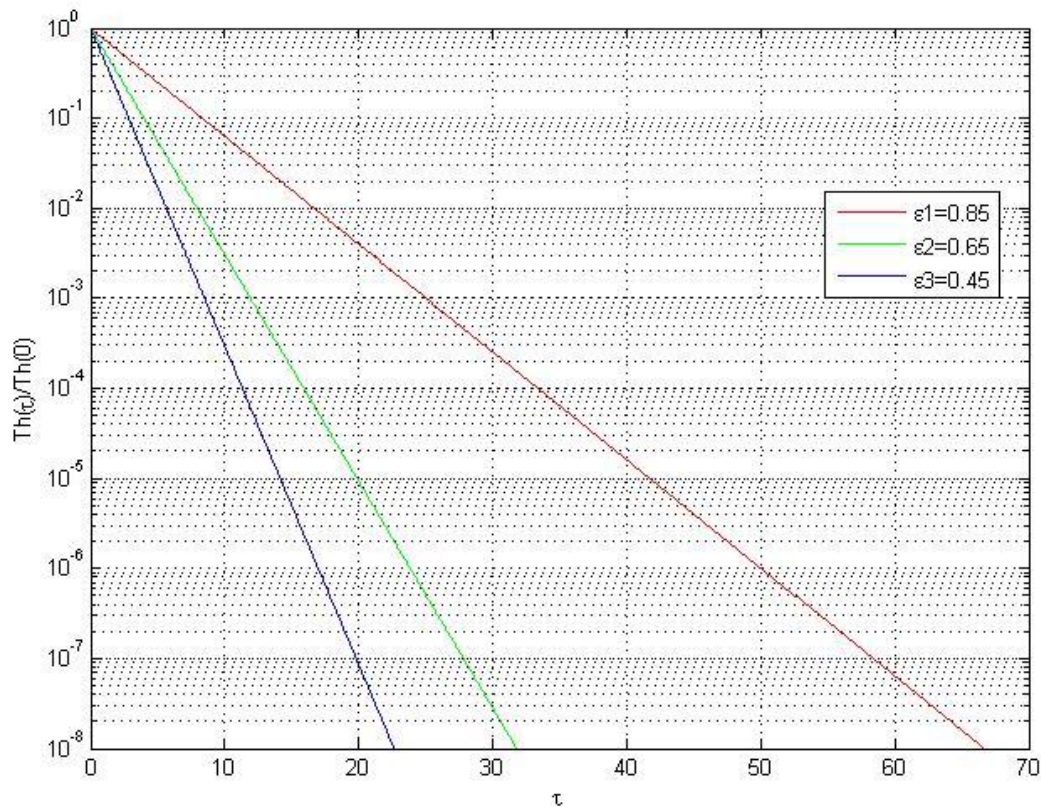
Први график (7.1) односи се на приказ Хафовог закона $T_H(\tau) = T_H(0) * \exp(2\gamma\tau)$ у $\log\text{-}\log$ скали, за три различите вредности коефицијента реституције: $\varepsilon_1 = 0,85$, $\varepsilon_2 = 0,65$, $\varepsilon_3 = 0,45$. Остале вредности параметара које су коришћене при рачунању коефицијената потребних за добијање графика су: $N = 2600$, $L = 2L_{box} = 240$, $\rho = 0,1$.



Слика 7.1 – график зависности $\frac{T_H(\tau)}{T_H(0)}$ од τ у $\log\text{-}\log$ скали за различите вредности ε

Приметимо још и да на основу једначине (6.3) следи: $\frac{T_H(\tau)}{T_H(0)} = \frac{E_k(\tau)}{E_k(0)}$.

Ову зависност можемо приказати и у семилогаритамској скали:

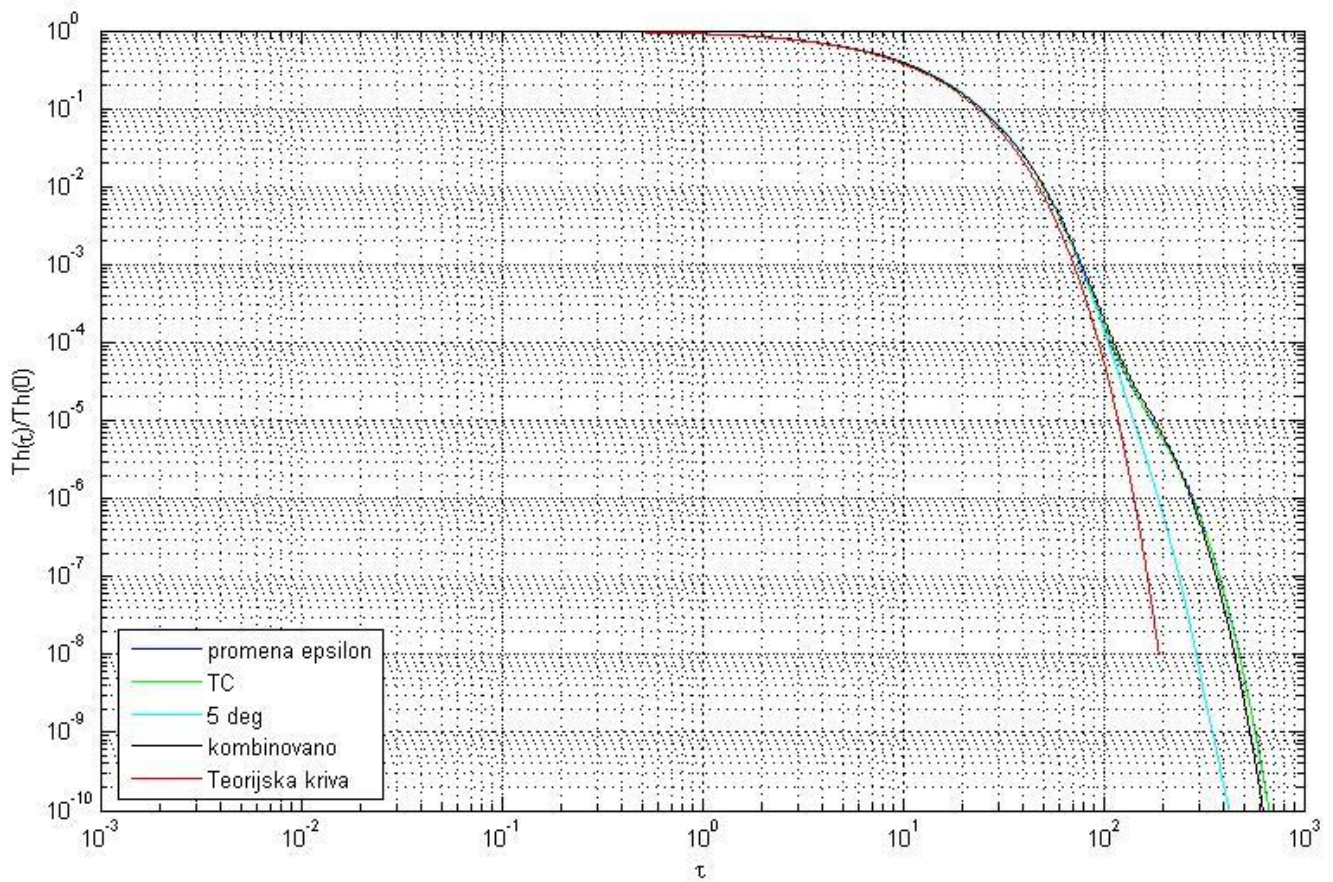


Слика 7.2 - график зависности $\frac{T_H(\tau)}{T_H(0)}$ од τ у семилогаритамској скали за различите вредности ε

Резултати нумеричких симулација хлађења грануларног гаса су директно поређени са теоријском кривом Хафовог закона. Тестирани су различити кодови, засновани на различитим методама избегавања нееластичног колапса. Ове методе су редом:

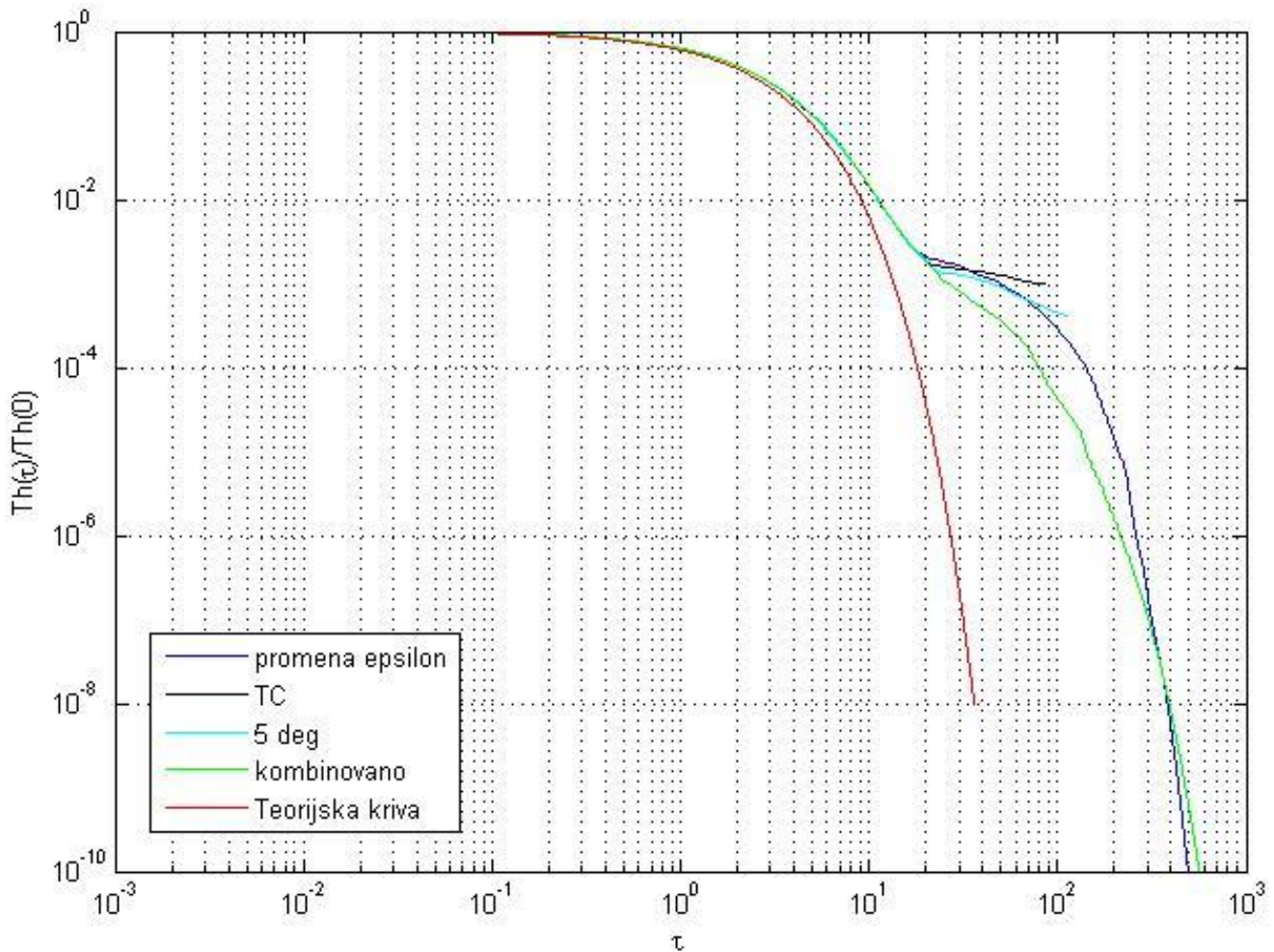
- 1) промена коефицијента реституције ε на фиксан број близак јединици када је релативна брзина судара довољно мала,
- 2) метод промене коефицијента реституције ако је време између два судара довољно мало,
- 3) ротација вектора брзина за неки произвољан угао из интервала ($0^\circ - 5^\circ$) и
- 4) комбинована метода промене коефицијента реституције када је релативна брзина мала и ротација вектора брзина за произвољан угао из интервала ($0^\circ - 5^\circ$).

Различите методе за избегавање нееластичног колапса су прво тестиране у симулацијама са следећим сетом параметара: број честица је био $N=2600$, коефицијент реституције $\epsilon=0,95$, димензија области $Lbox=120$. Ово су три независна параметра (уз фиксан параметар - полупречник честица, који је узет за 1). Међутим да би имали бољи преглед резултата можемо користити и густину система честица. У овом случају густина система би била $\rho = \frac{N \cdot d^2 \cdot \pi}{4 \cdot Lbox^2} = 0,142$. График зависности нормиране кинетичке енергије система од времена у новој временској скали је дат на слици (7.3).



Слика 7.3 – график зависности нормиране кинетичке температуре од времена τ у новој временској скали за $\epsilon=0,95$

Различите методе за избегавање нееластичног колапса потом су тестиране са другим сетом параметара: $N=2600$, коефицијент реституције $\epsilon=0,70$, димензија области $L_{box}=120$, и такође $\rho = 0,142$. График је дат на слици (7.4).



Слика 7.3 – график зависности нормиране кинетичке температуре од времена τ у новој временској скали за $\epsilon=0,70$

Примећујемо да боље слагање са Хафовим законом дају симулације са већим коефицијентом реституције. То је и очекивано, јер систем са већим коефицијентом реституције дуже остаје у хомогеном стању. Пошто уочавамо релативно мала одступања резултата симулација, ми одабирамо једну за даља испитивања и то ће бити комбиновани метод.

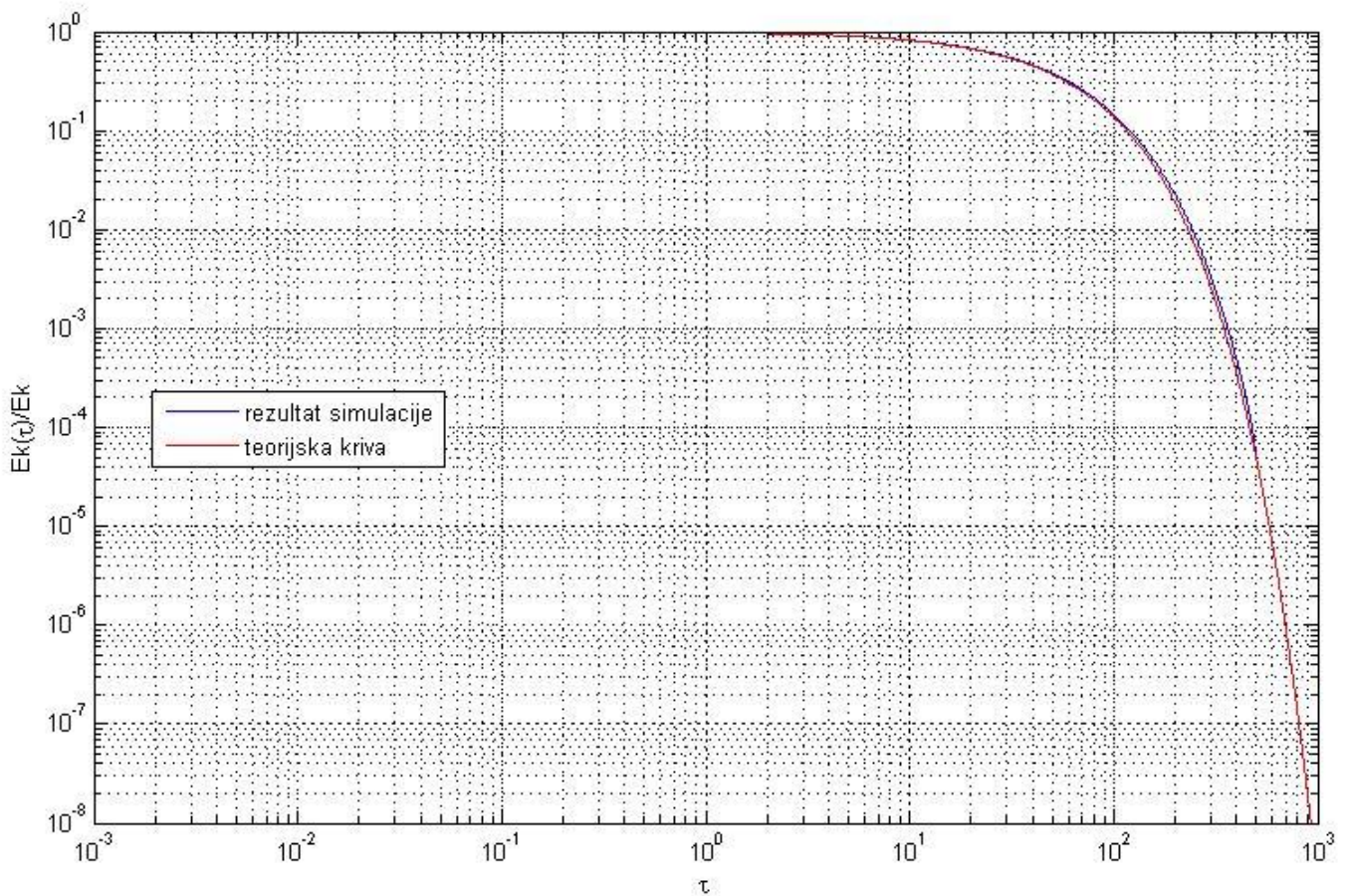
8 Динамика хлађења грануларног гаса у две димензије

У овом поглављу ћемо испитивати појаву одређених уређености како у просторном распореду честица тако и од брзине. У раду „*Dynamics of a freely evolving, two-dimensional granular medium*“ (McNamara и Young, 1996 [3]) примећено је да се, зависно од коефицијента реституције, јављају различите „фазе“ током хлађења грануларног гаса. Ове фазе су назване: кинетичка, фаза смицања и фаза кластеровања.

При приказивању свих ових фаза користили смо углавном следеће вредности за број честица и димензију области, односно густину: $N = 2600$, $L = 2L_{box} = 240$, $\rho = 0,142$. Промена се састојала у различитим вредностима ε . Наиме, McNamara и Young су показали да када је коефицијент реституције довољно близак јединици, односно $0,98 < \varepsilon < 1$, за густину система од $\rho = 0,25$ и број честица $N = 1024$, тада систем довољно дуго остаје хомоген [3]. Ако смањујемо коефицијент реституције током еволуције система долази до појаве уређености. Тако имамо да за $0,83 \leq \varepsilon \leq 0,97$, при истим вредностима ρ и N , долази до појаве одређених токова честица након одређеног броја судара по честици (времена τ_{sc}). Примећујемо, дакле, да долази до уређености импулса. Ако даље смањујемо коефицијент реституције, долази до појаве нехомогености густине броја честица, тј. до гомилања честица у одређеним деловима простора. Те нехомогености се називају кластери и њихова појава је стохастичке природе.

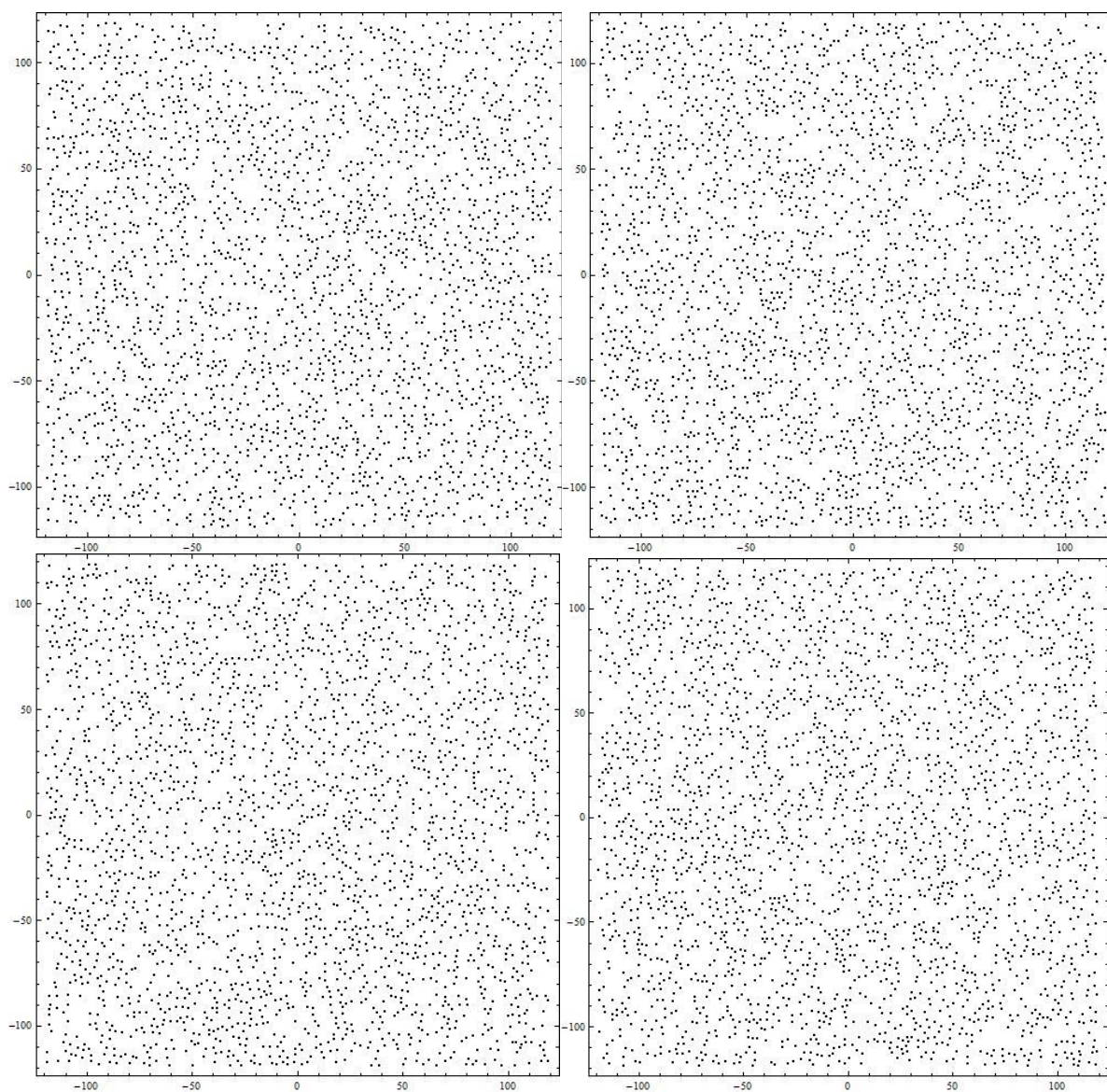
8.1 Кинетичка фаза

Кинетичку фазу хлађења грануларног гаса приказујемо за систем са коефицијентом реституције $\varepsilon = 0,99$. Систем је веома дуго остао у хомогеној фази што смо приказали на сликама 8.2 и 8.3. На слици 8.1 је дат приказ нормиране средње кинетичке енергије система (која је еквивалентна нормираној кинетичкој температури) од времена τ . Друга крива на слици 8.1 је теоријска крива и примећујемо да док је год систем хомоген, резултати симулације прате теоријску криву.



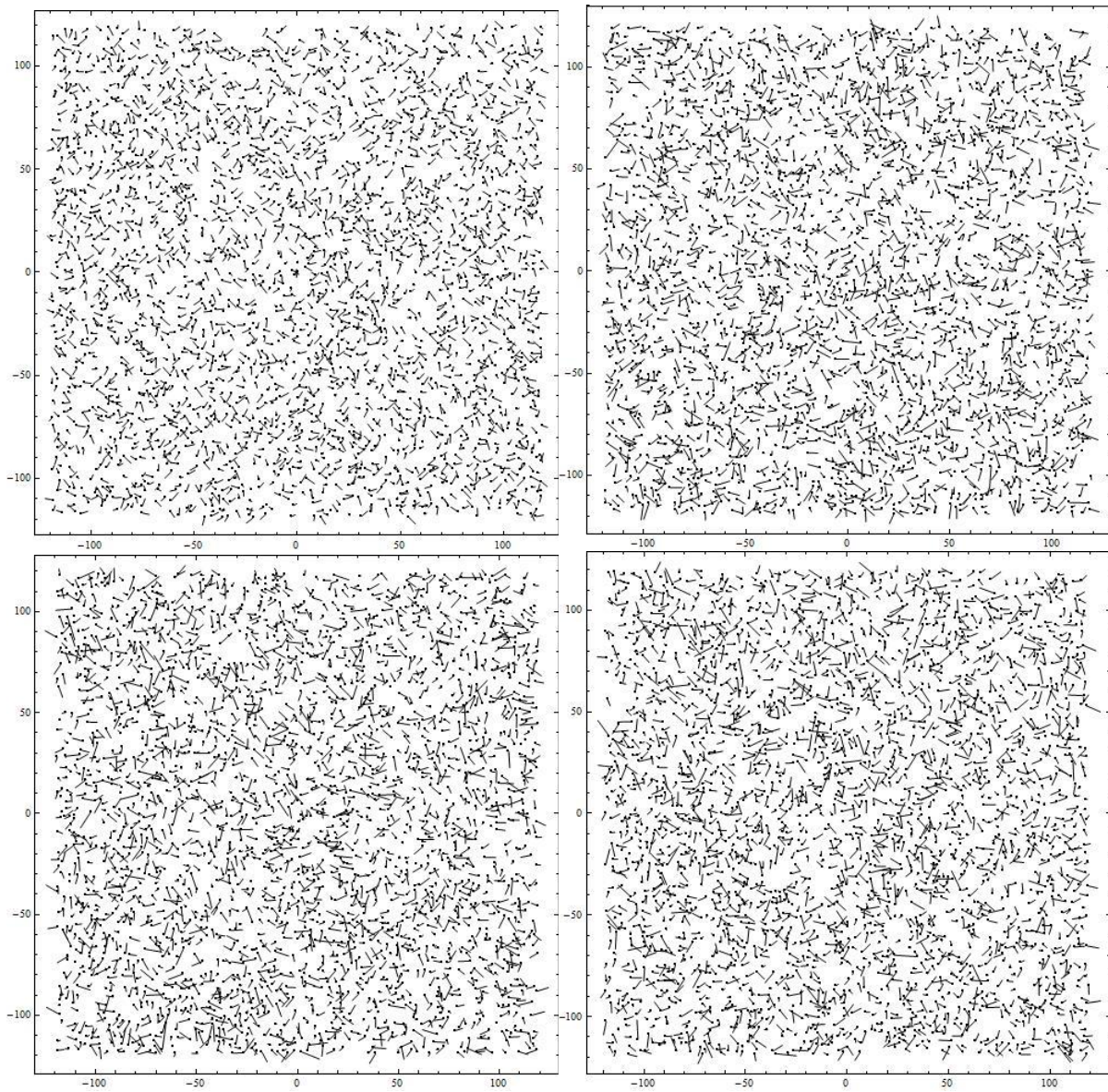
слика 8.1 – поређење резултата симулације са теоријском кривом Хафовог закона, за $\varepsilon = 0,99$

На слици 8.2 приказане су честице у почетном стању и након: $\tau = 38,46$, $\tau = 192,3$, $\tau = 461,54$. Видимо да у овом времену не долази до појаве нехомогености.



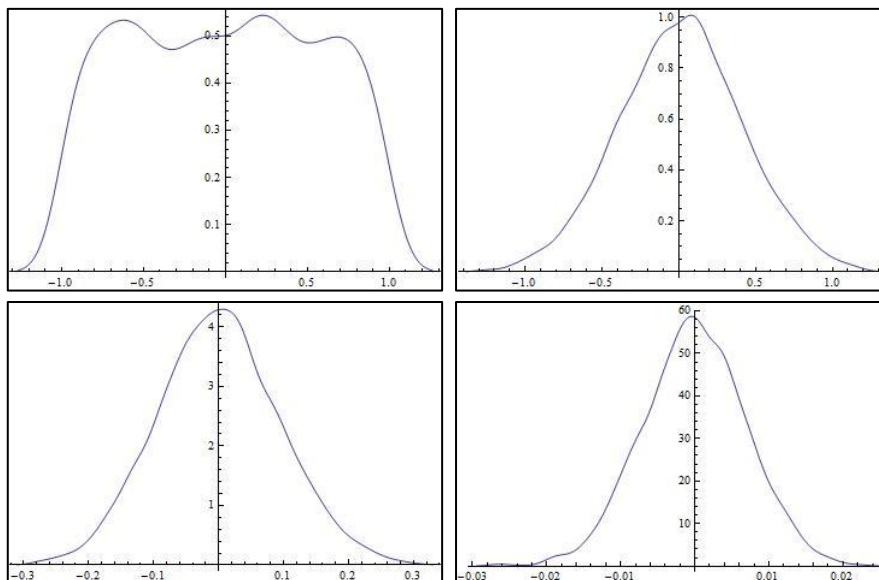
Слика 8.2 – приказ кинетичке фазе у почетном стању (горе лево) и након: $\tau = 38,46$ (горе десно), $\tau = 192,3$ (доле лево), $\tau = 461,54$ (доле десно)

У програму *Mathematica* је урађен и приказ честица са векторима брзина (где дужина линије представља релативни интензитет брзине). На слици 8.3 се може видети да није дошло ни до формирања уређености брзина.

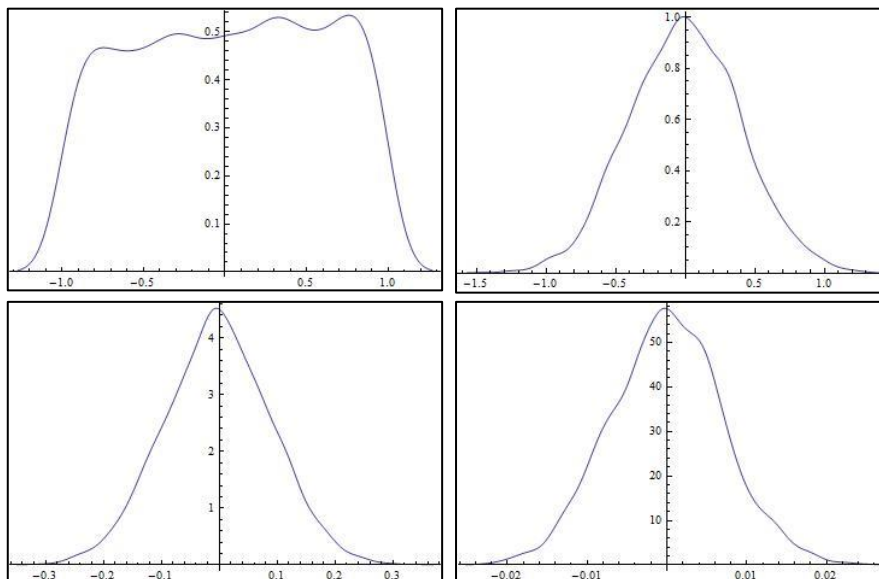


Слика 8.3 – приказ честица са векторима брзина у почетном стању (горе лево) и након: $\tau = 38,46$ (горе десно), $\tau = 192,3$ (доле лево), $\tau = 461,54$ (доле десно)

На сликама 8.4 и 8.5 су приказани хистограми по компонентама брзина за исте случајеве који су наведени и на сликама 8.2 и 8.3. Видимо да је у почетку релативно хомоген распоред брзина из интервала $(-1, 1)$ који потом прелази у распоред сличан равнотежној дистрибуцији по брзинама. Приметимо и да се дисперзија брзина смањивала током времена.



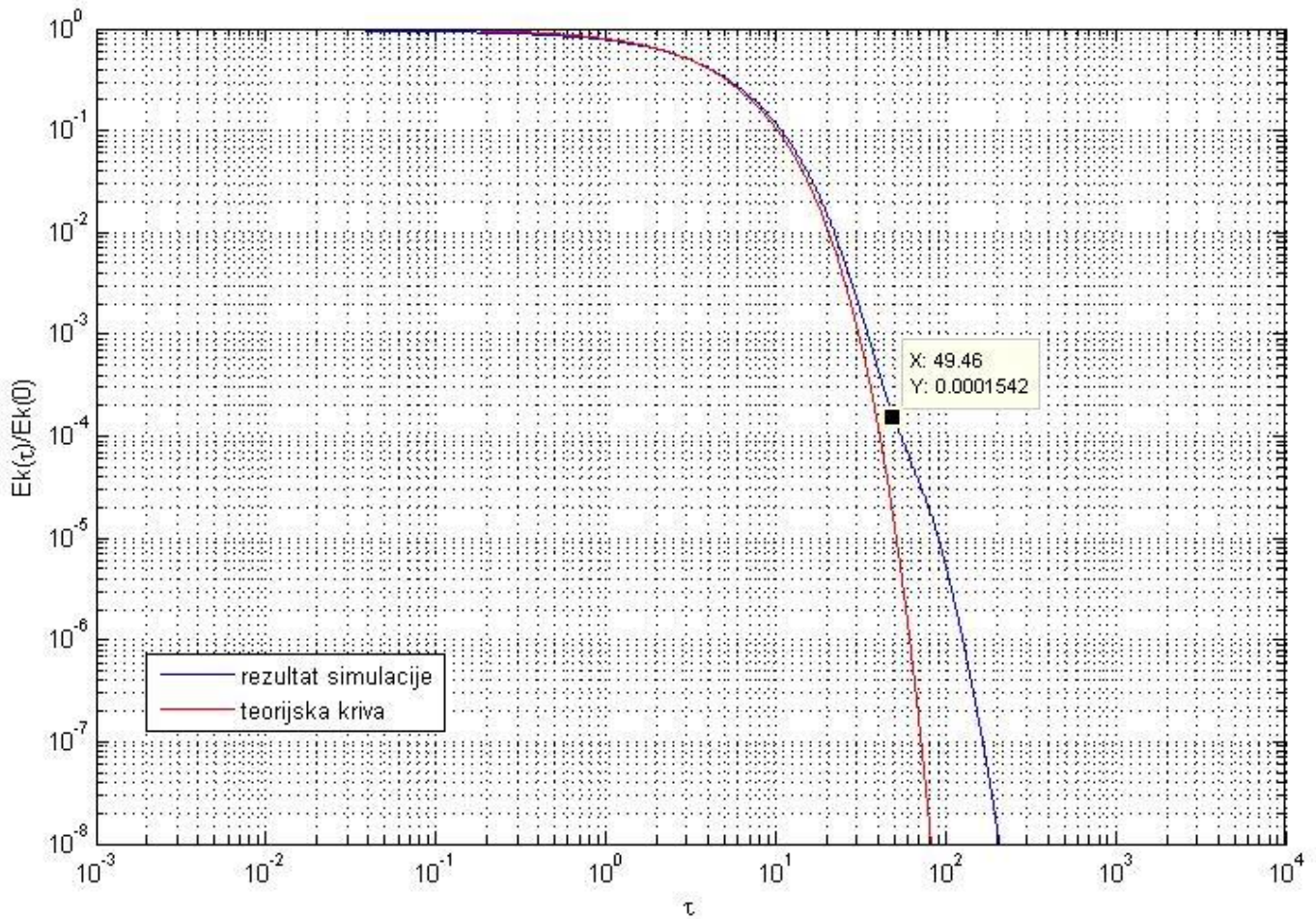
Слика 8.4 – хистограм v_x компоненте у почетном тренутку (горе лево) и након: $\tau = 38,46$ (горе десно), $\tau = 192,3$ (доле лево), $\tau = 461,54$ (доле десно)



Слика 8.5 – хистограм v_y компоненте у почетном тренутку (горе лево) и након: $\tau = 38,46$ (горе десно), $\tau = 192,3$ (доле лево), $\tau = 461,54$ (доле десно)

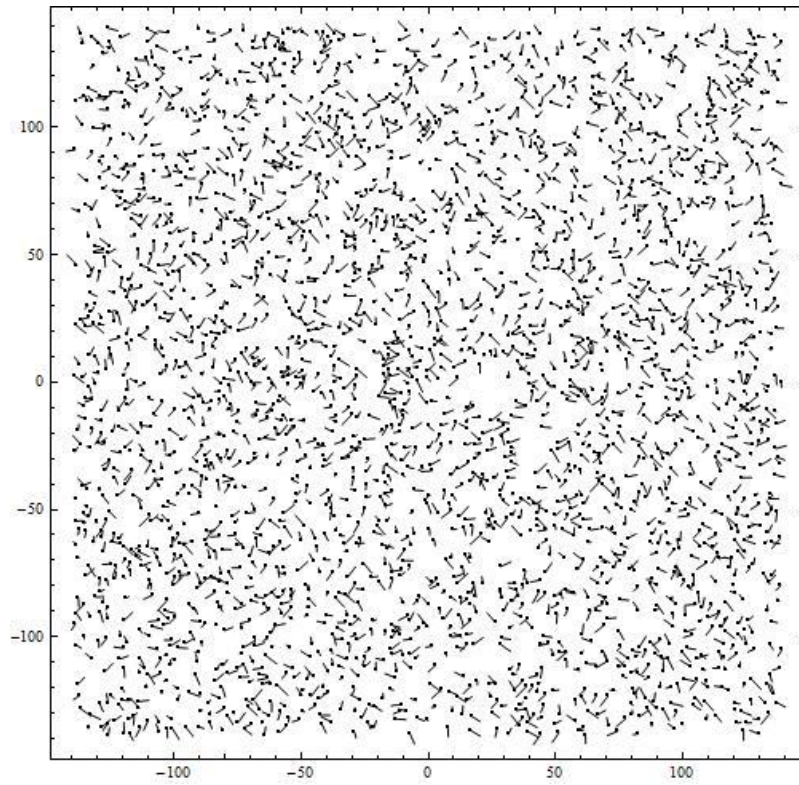
8.2 Фаза смицања

У овом делу презентујемо податке за следеће параметре у симулацији: $\varepsilon = 0,88$, $N = 2600$, $L = 2L_{box} = 280$, $\rho = 0,104$. Примећено је да долази до формирања одређених структура, пре свега у виду токова честица. Хомогена фаза се брзо нарушава ($\tau \cong 50$) и тада почиње веће неслагање са Хафовим законом (слика 8.6).

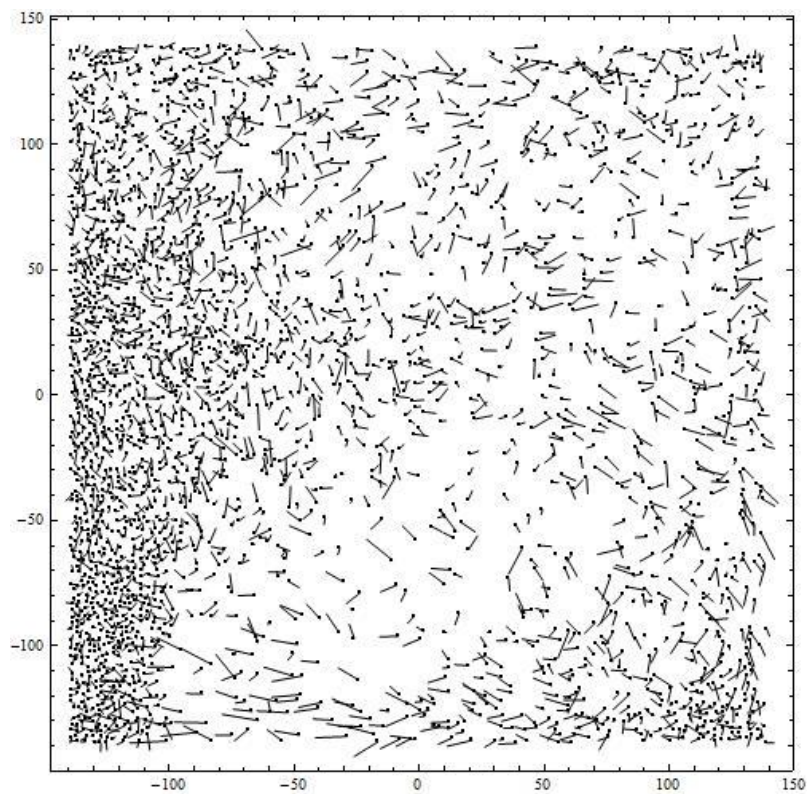


Слика 8.6 - поређење резултата симулације са теоријском кривом Хафовог закона, за $\varepsilon = 0,88$

Такође, можемо приказати резултате симулација за различита времена, у почетку, након прве појаве нехомогености и при оформљеним токовима честица. На сликама 8.8, 8.9 и 8.10 видимо да се токови прво јављају негде на средини области и ту се деле на два тока. Потом долази до њихове деформације, и на крају прелазе у један ток, односно кружно кретање (слика 8.11).

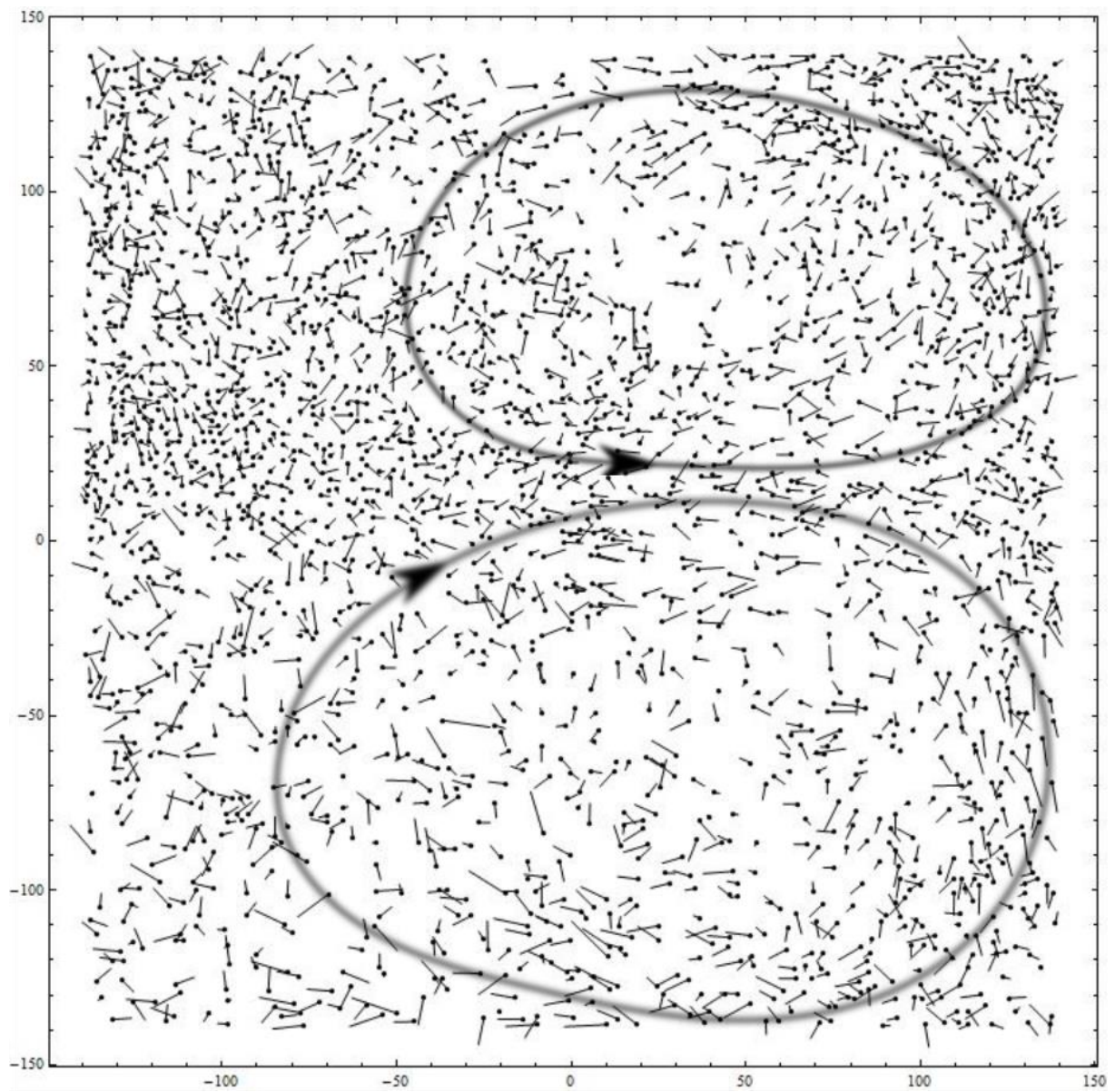


Слика 8.7 - приказ честица са векторима брзина у почетном стању



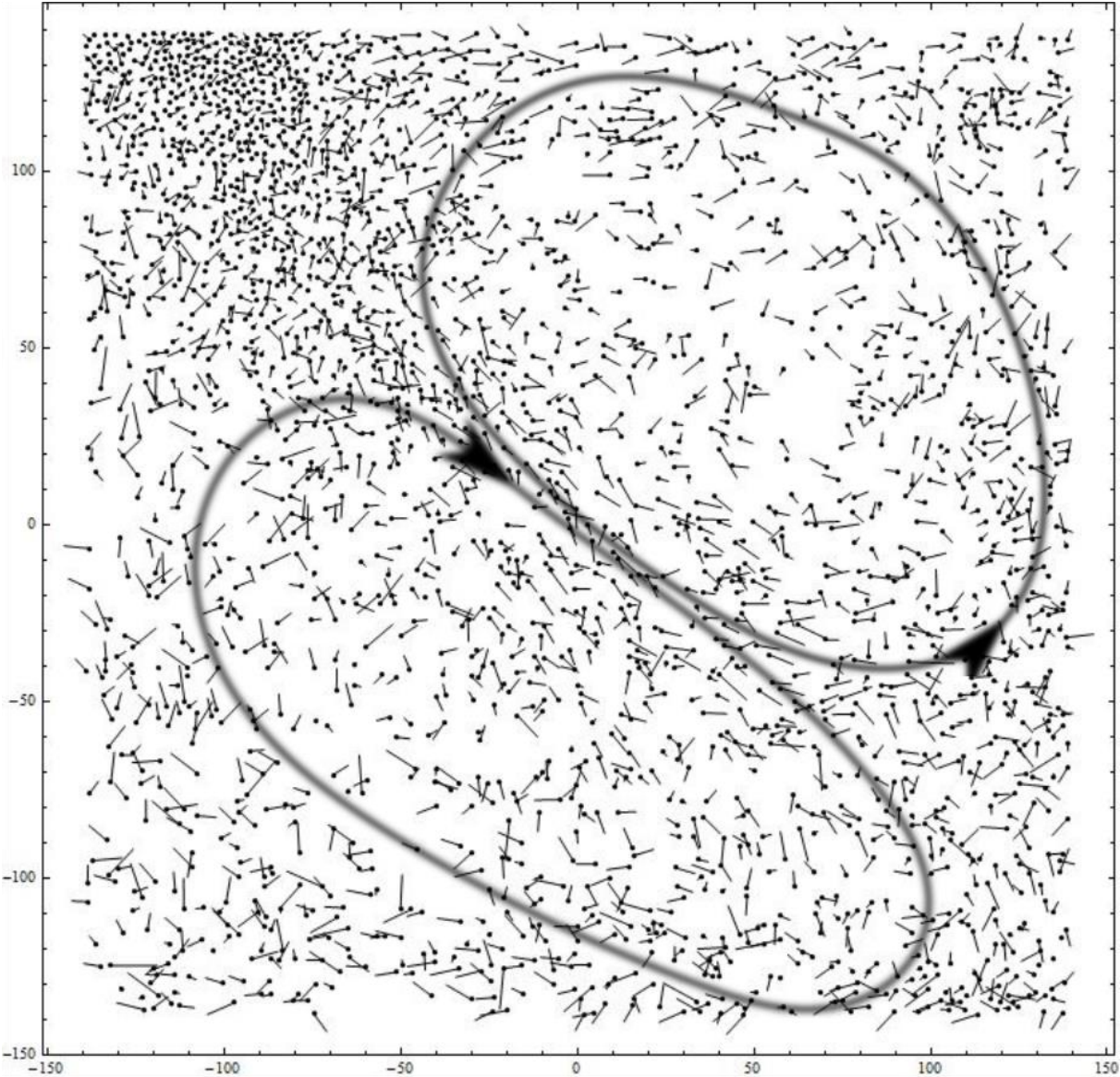
Слика 8.8 - приказ честица са векторима брзина за $\tau = 50,00$

На слици 8.9 можемо видети формирање токова честица у тренутку $\tau = 119,23$.



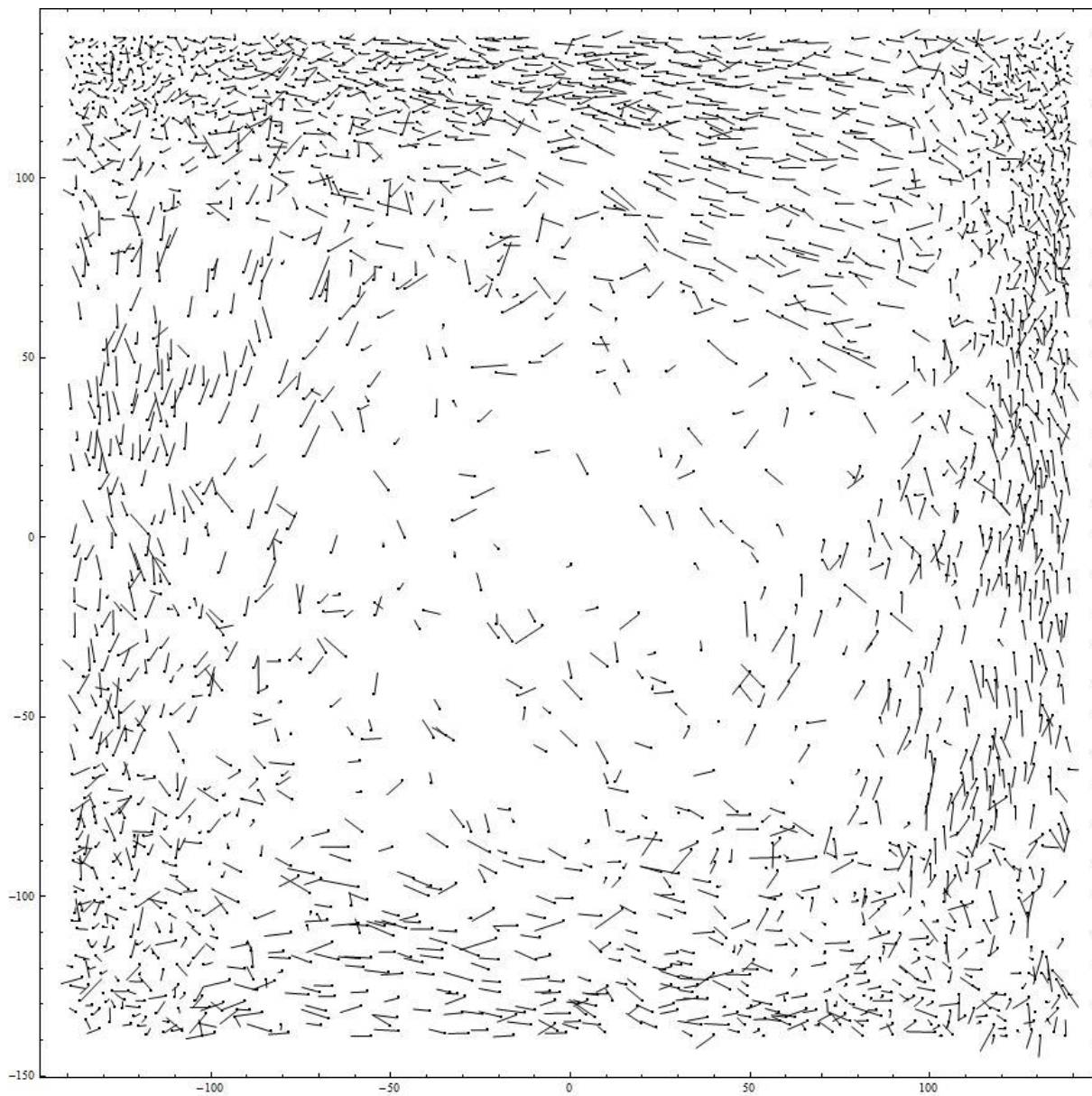
Слика 8.9 - приказ токова честица за $\tau = 119,23$

Потом се тај ток деформише:



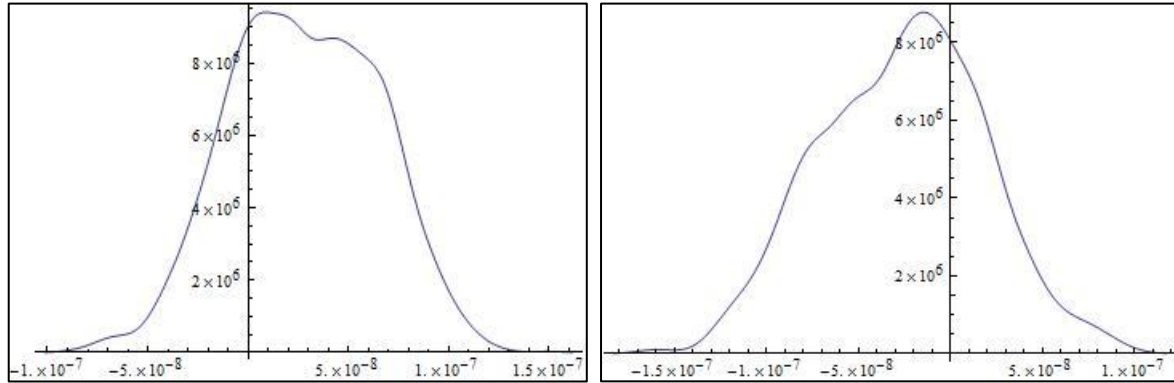
Слика 8.10 - приказ токова честица за $\tau = 165,38$

И на крају долази до формирања једног тока:

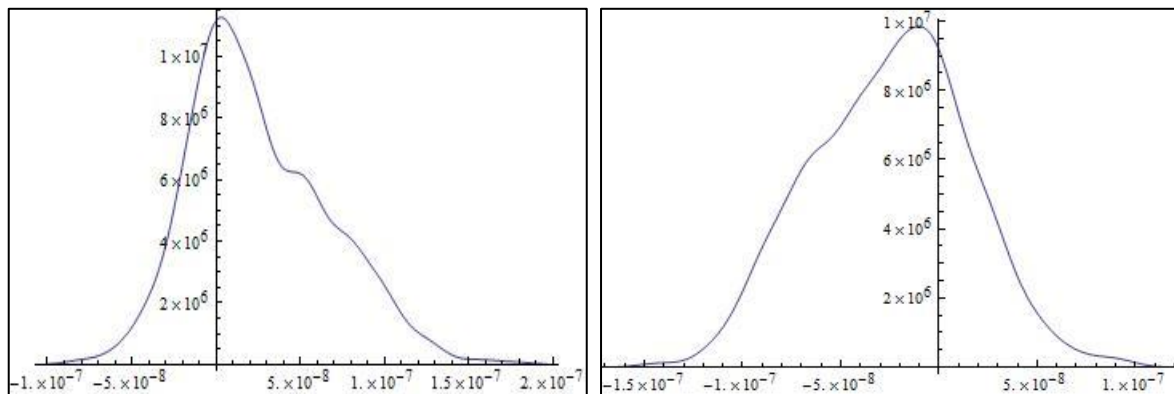


Слика 8.11 - приказ тока честица за $\tau = 538,5$

Можемо и графички приказати уређеност брзина, тиме што ћемо поделити област на две половине. Ако посматрамо честице са координатама $y > 0$, а потом $y < 0$, можемо приметити уређеност по x -компоненти брзине. Аналогно, ако посматрамо честице подељене на оне са $x > 0$ и $x < 0$ тада имамо уређеност по y -компоненти брзине (слике 8.12 и 8.13).



Слика 8.12 – хистограми v_x компоненти брзине за области $y \geq 0$ (лево) и $y \leq 0$ (десно) у тренутку $\tau = 538,5$

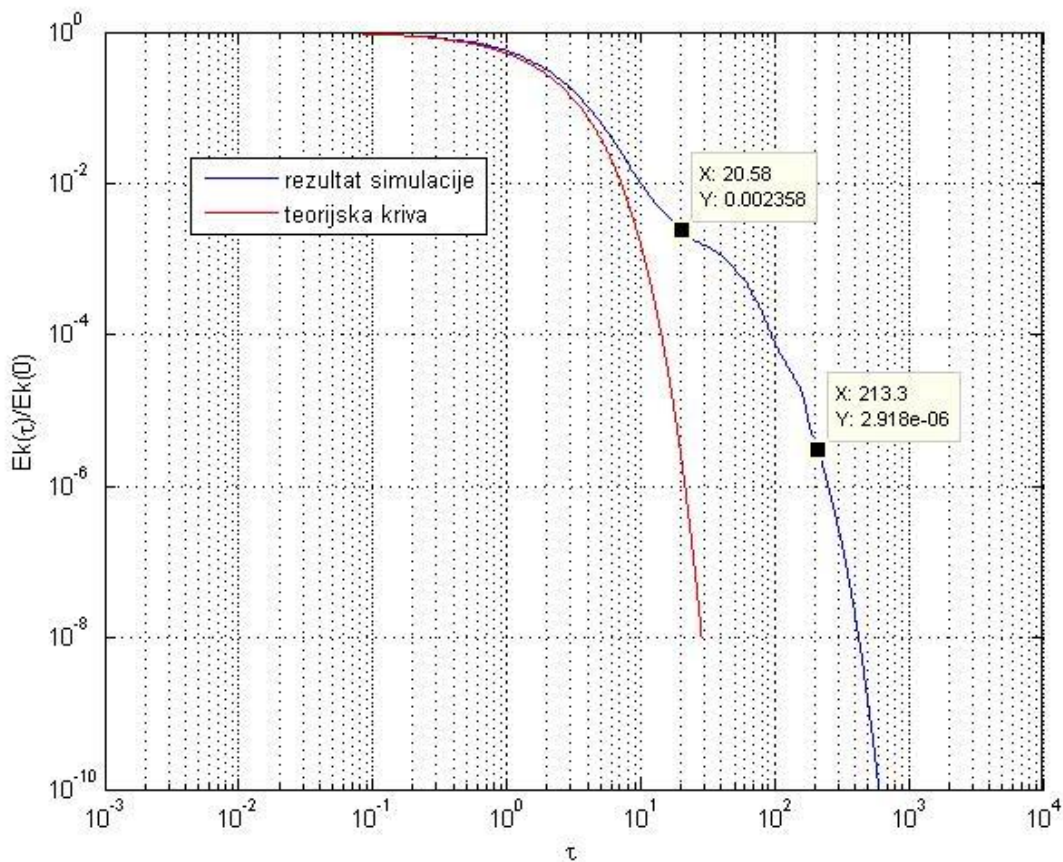


Слика 8.13 – хистограми v_y компоненти брзине за области $x \leq 0$ (лево) и $x \geq 0$ (десно) у тренутку $\tau = 538,5$

8.3 Фаза кластеровања

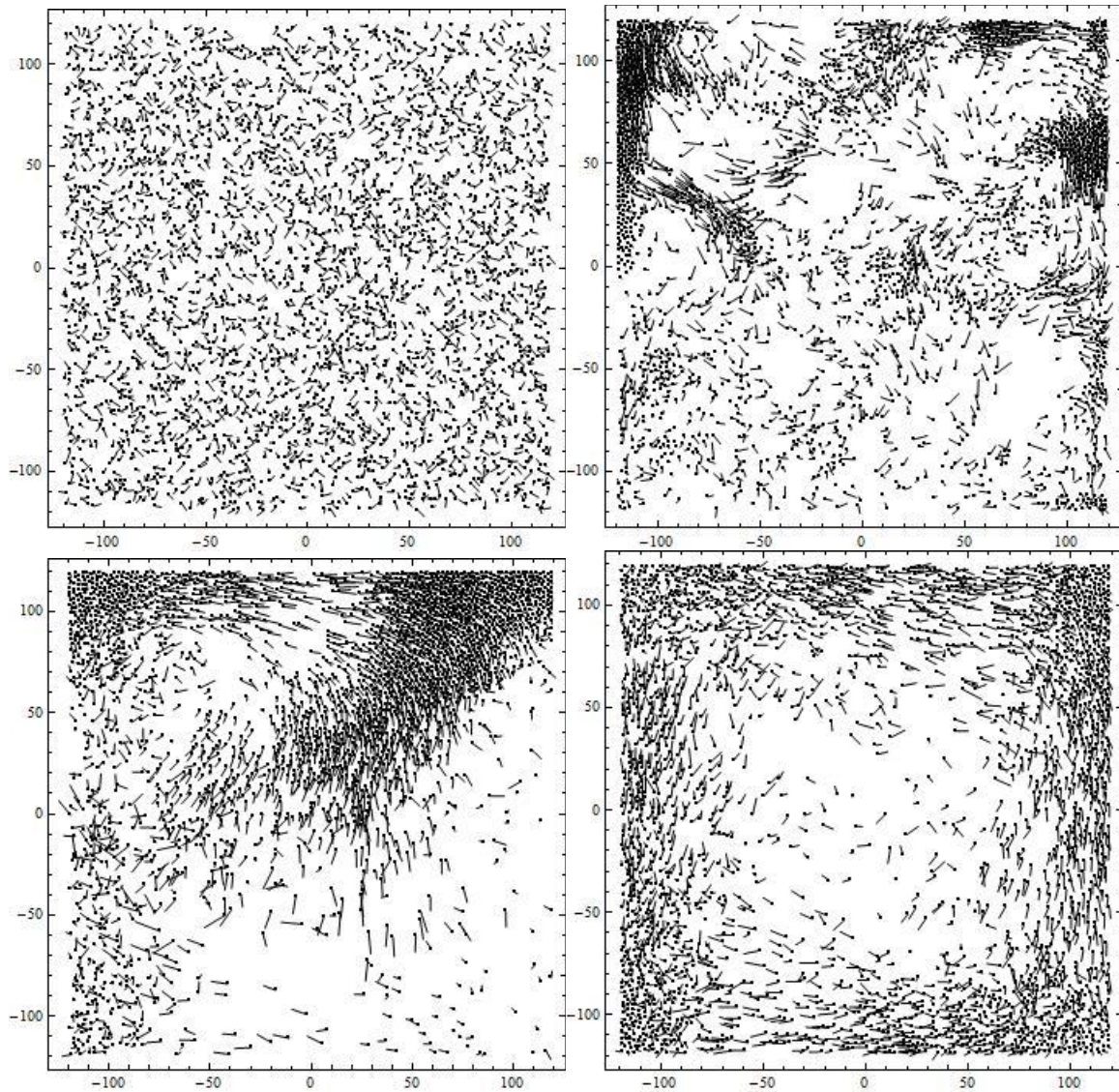
У раду „Clustering Instability in Dissipative Gases“ [15] разматран је проблем формирања кластера у процесу хлађења грануларног гаса. Наиме, ако услед стохастичке природе процеса у неком делу запремине грануларног гаса порасте густина честица, повећа се и број судара у том делу. Пошто је судар нееластичан, грануларна температура у тој области ће почети да опада брже него у осталим деловима запремине, смањујући притисак. Смањење притиска за последицу има миграцију осталих честица ка тој области и даље повећање њене густине. Дакле, једном формирана, неравномерност у густини индукује формирање кластера. При том је потребно и да процеси који потичу од хидродинамичких механизма који теже да онемогуће процес кластеровања буду спорији од процеса кластеровања

Да би приказали процес кластеровања, ми смо користили следеће вредности параметара: $N = 2600$, $L = 2L_{box} = 240$, $\rho = 0,142$ и $\varepsilon = 0,60$. Примећено је да се кластери формирају врло брзо, након $\tau \cong 20$. Потом систем остаје у овој фази неко време, да би се касније јавиле и уређености у брзини ($\tau \cong 200$) које су примећене и у фази смицања. Оно што је карактеристично за овако мале коефицијенте реституције је да врло брзо почиње да се јавља одступање од Хафовог закона (слика 8.14).



Слика 8.14 - поређење резултата симулације са теоријском кривом Хафовог закона, за $\varepsilon = 0,60$

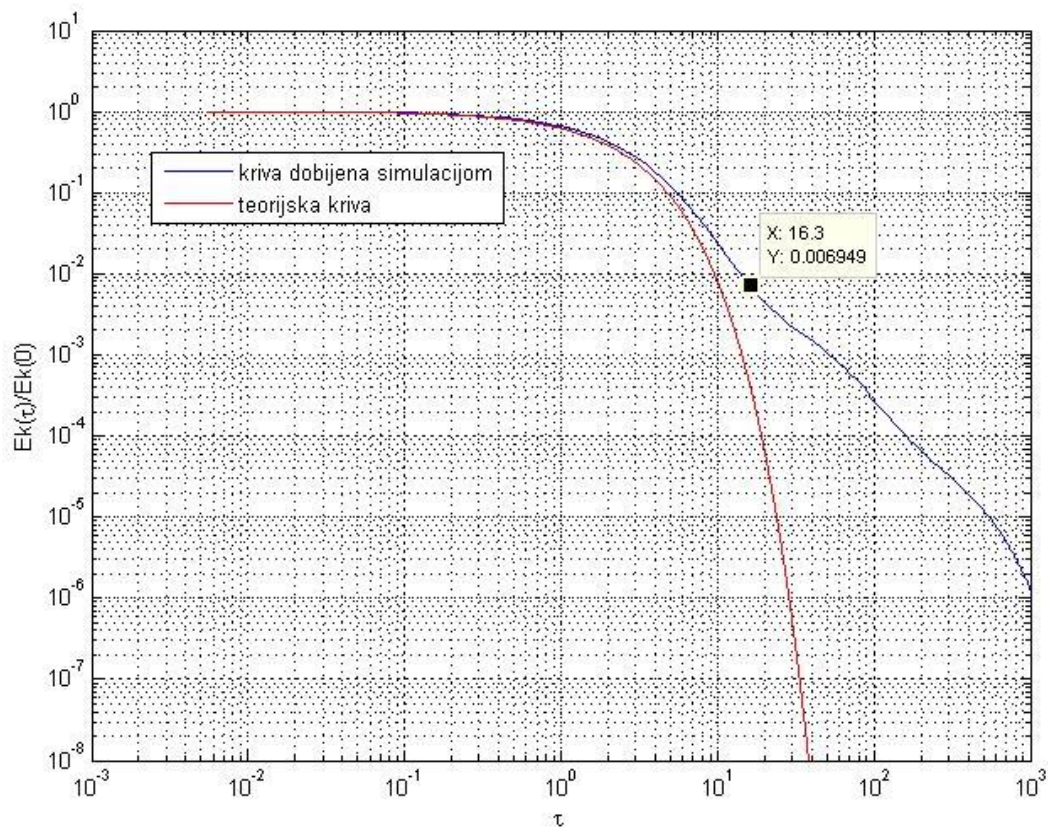
На претходној слици су приказане тачке (апроксимативно) између којих је систем у стању израженог кластерованја. Видимо доста неправилан ток криве добијене симулацијом у том периоду. Након времена $\tau \cong 200$ долази до правилнијег пада криве добијене симулацијом, и тада је примећено и формирање токова честица. Ово се јасно може видети на слици 8.15 где су приказана стања система у тренуцима $\tau = 0$, $\tau = 23,08$, $\tau = 138,46$, $\tau = 384,62$.



Слика 8.15 – приказ честица са векторима брзина у почетном стању (горе лево) и након: $\tau = 23,08$ (горе десно), $\tau = 138,46$ (доле лево), $\tau = 384,62$ (доле десно)

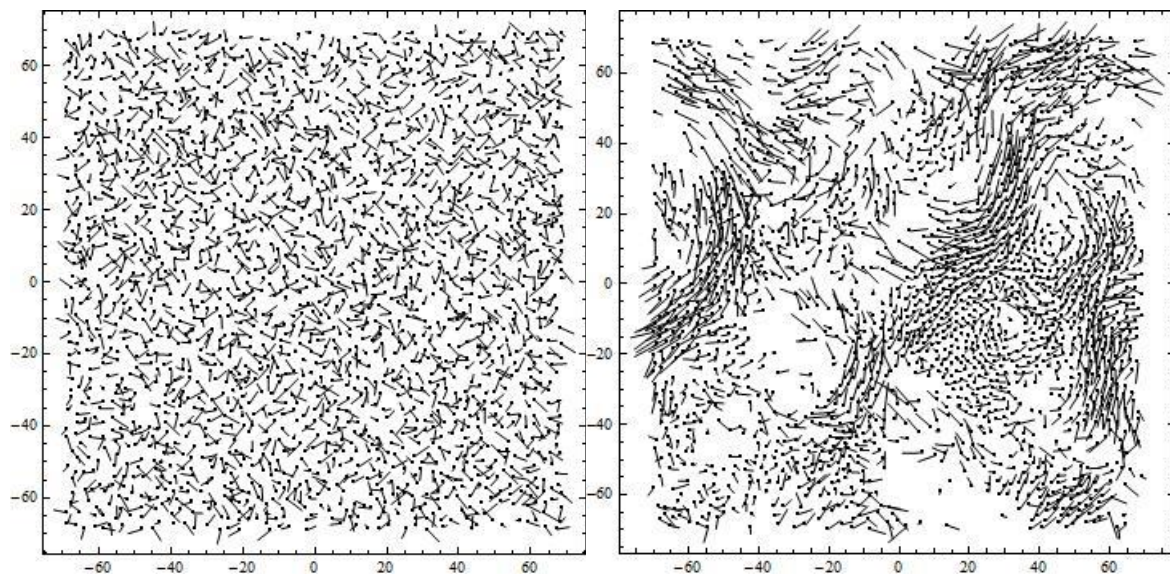
8.4 Периодични гранични услови

У овом делу рада биће анализирана кинетика дводимензионалног грануларног гаса који се налази у области са периодичним граничним условима у оба правца. У овим симулацијама коришћени су следећи параметри: $\varepsilon = 0,72$, $N = 2000$, $L = 2L_{box} = 140$, $\varrho = 0,32$. Пошто је ε било довољно мало очекивана је појава кластера што смо и запазили. Прво дајемо графике слагања теоријске и криве добијене симулацијом (слика 8.16).

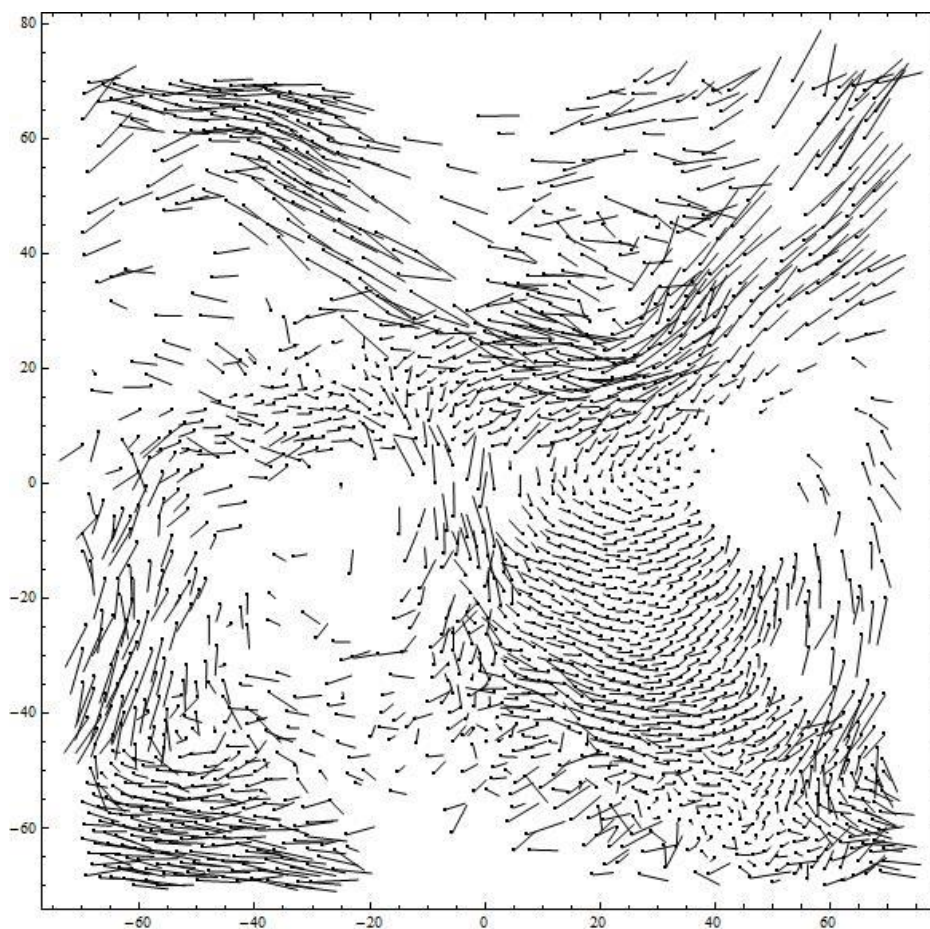


Слика 8.16 - поређење резултата симулације са теоријском кривом Хафовог закона, за $\varepsilon = 0,72$, периодични гранични услови

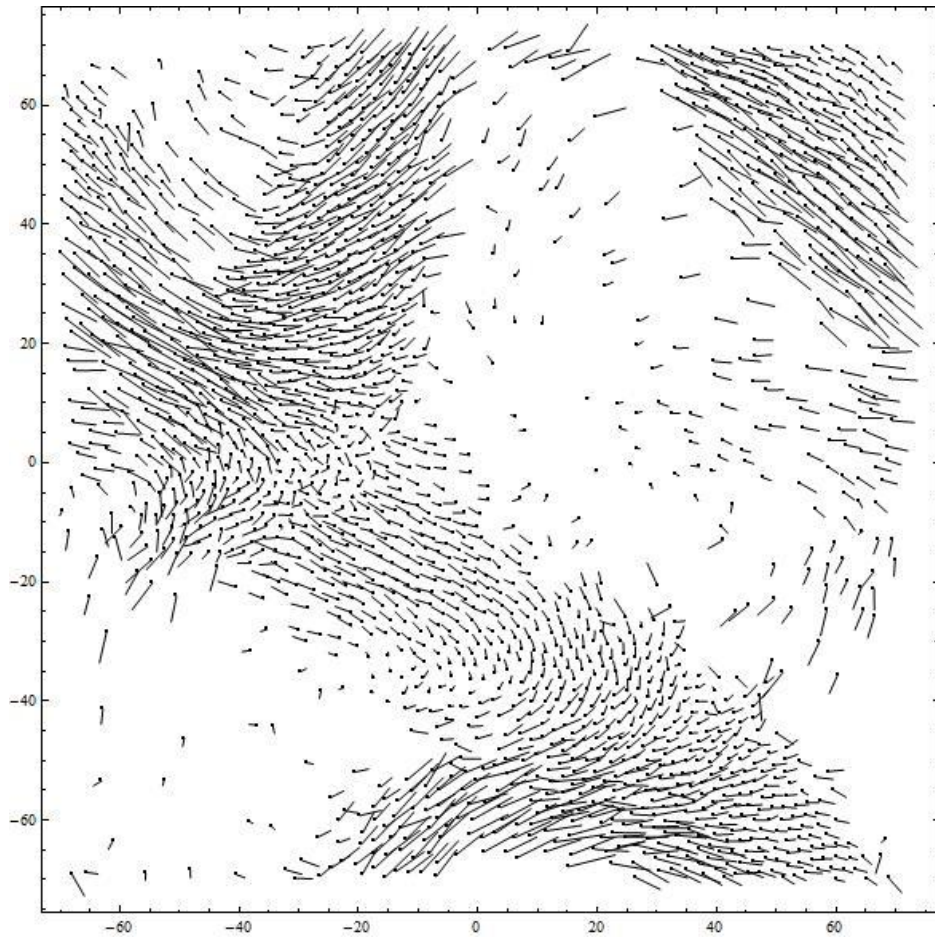
Уочавамо тачку ($\tau \cong 20$) код које почиње значајније одступање од Хафовог закона, и то је период када почиње фомирање кластера. Карактеристична стања система приказана су на сликама (8.17 – 8.19).



Слика 8.17 – приказ честица са векторима брзина у почетном стању (лево) и након: $\tau = 20$ (десно)



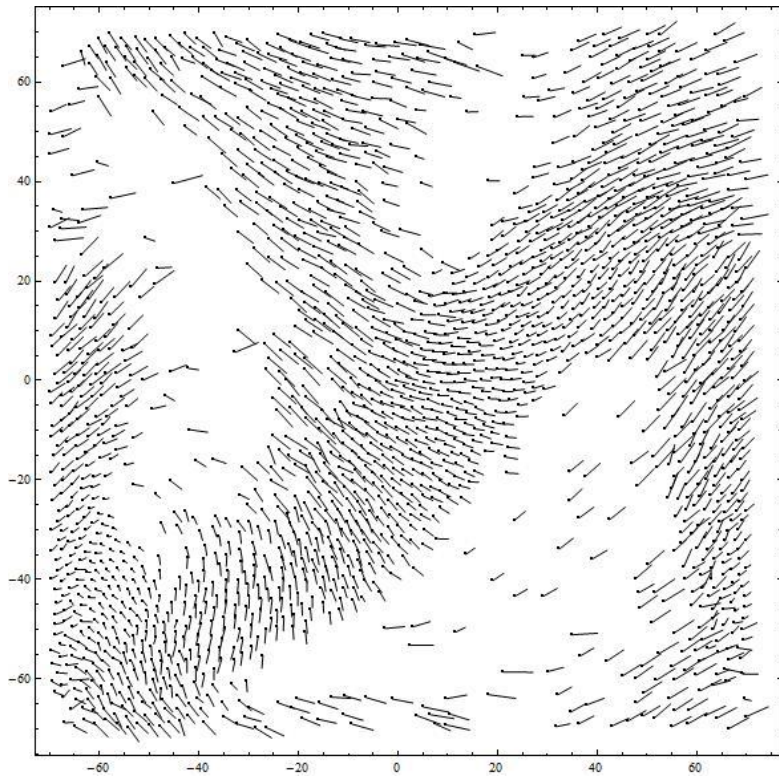
Слика 8.18 – приказ честица са векторима брзина након: $\tau = 50$



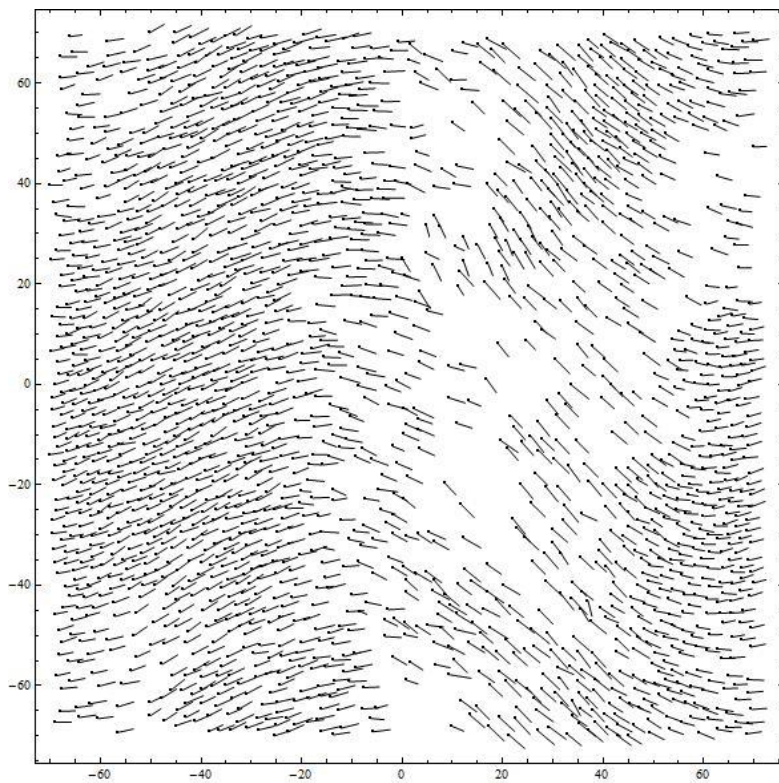
Слика 8.19 – приказ честица са векторима брзина након: $\tau = 100$

Након тога, као и код фазе кластеровања за чврсте зидове, примећујемо да просторна нехомогеност почиње да се смањује. Истовремено се стварају макроскопски токови честица.

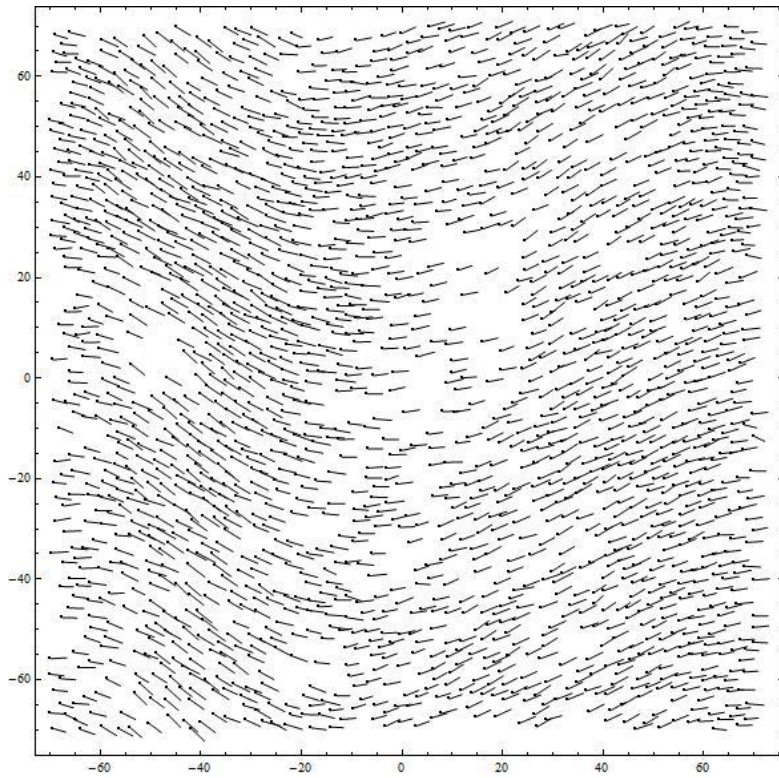
У времену $\tau = 150$ примећујемо два делимично независна тока (слика 8.20), која су добијена и у раду [3].



Слика 8.20 – приказ честица са векторима брзина након: $\tau = 150$

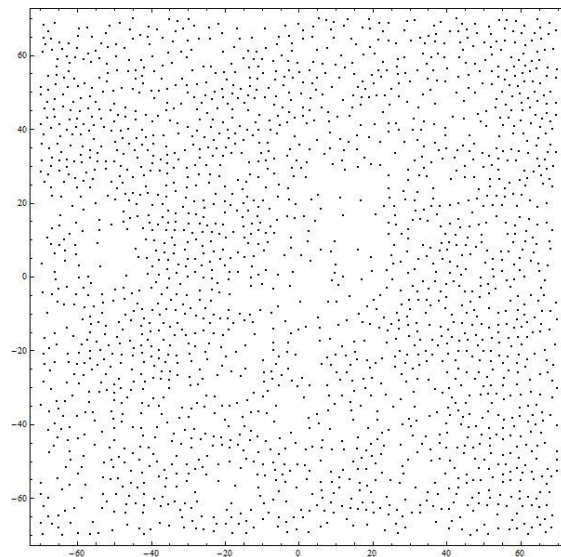


Слика 8.21 – приказ честица са векторима брзина након: $\tau = 225$



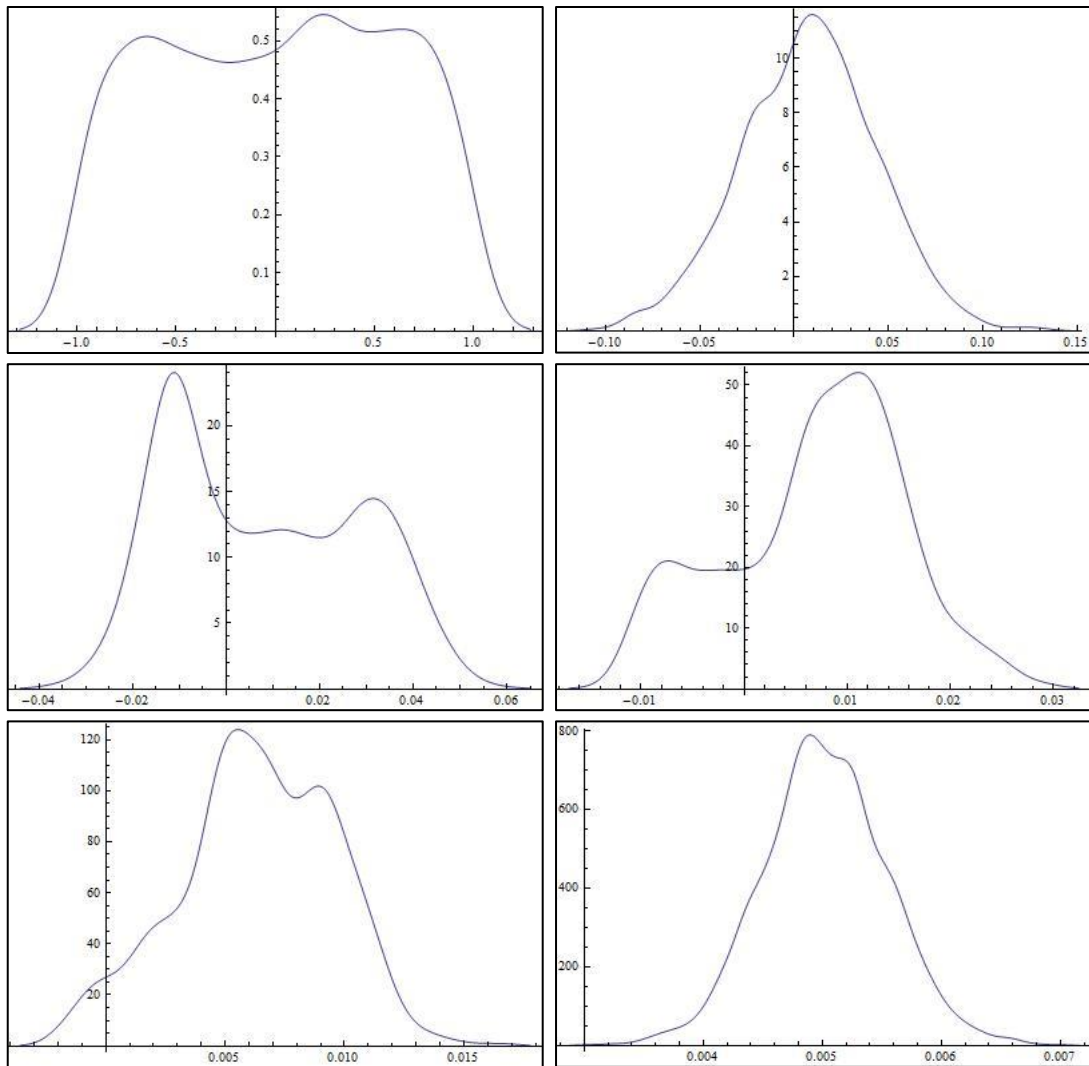
слика 8.22 – приказ честица са брзинама у тренутку $\tau = 250$

Након времена $\tau = 250$ формира се релативно стабилна конфигурација (стационарно стање). Примећујемо да се просторна нехомогеност скоро потпуно изгубила (слика 8.23).

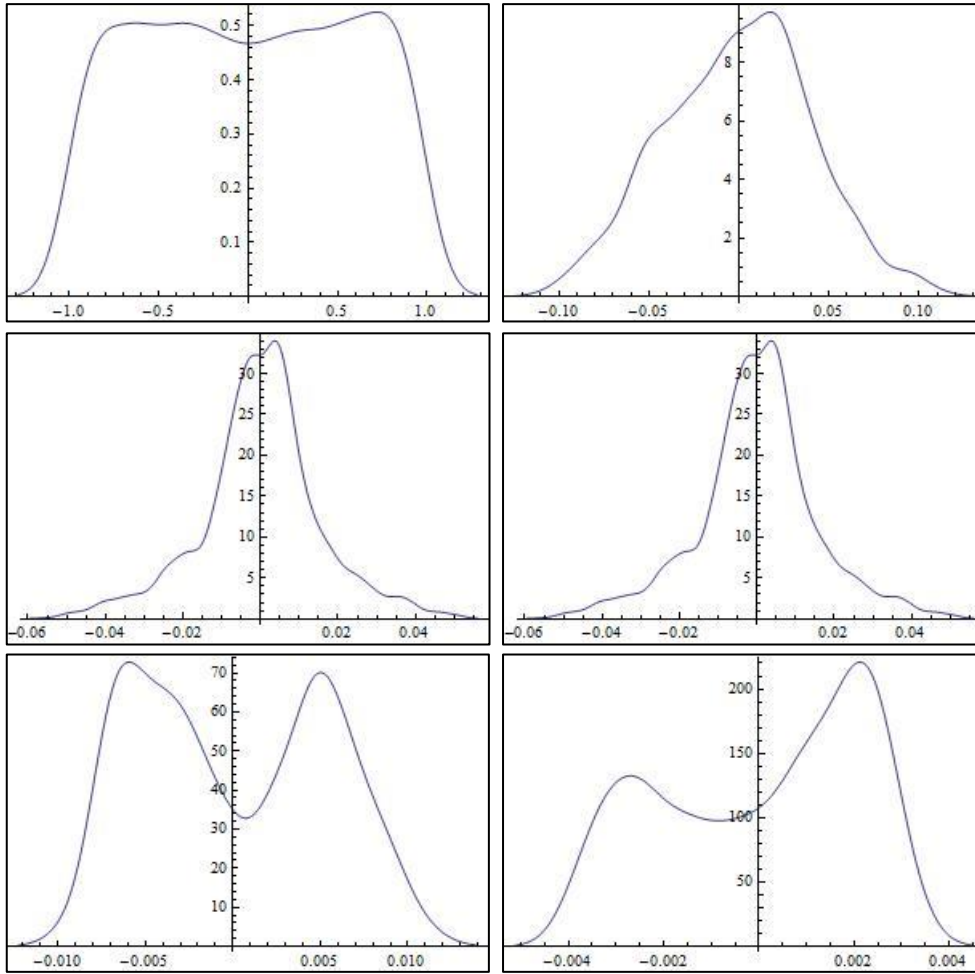


Слика 8.23 – приказ честица у тренутку $\tau = 250$

Осим визуелно, постојање макроскопских токова честица може се потврдити и коришћењем хистограма компоненти брзина честица (слике 8.24 и 8.25). Примећујемо делимично независне токове за времена $\tau \cong (50, 250)$



Слика 8.24 - хистограми v_x компоненти брзине у тренуцима: почетни тренутак, $\tau = 20$ (први ред), $\tau = 50$, $\tau = 100$ (други ред), $\tau = 150$, $\tau = 250$ (трећи ред)



Слика 8.25 - хистограми v_y компоненти брзине у тренуцима: почетни тренутак, $\tau = 20$ (први ред), $\tau = 50$, $\tau = 100$ (други ред), $\tau = 150$, $\tau = 250$ (трећи ред)

9 Закључак

У овом раду смо приказали еволуцију разређеног грануларног система помоћу нумеричких симулација. У њима су коришћене апроксимације о сферним гранулама једнаких радијуса, потом о инфинитезимално малој површини судара, нисмо узимали у обзир силе трења, нити смо разматрали ротациона кретања. Све симулације су вршене у две димензије, без гравитације.

И поред тога видели смо битне феномене који се дешавају и у реалним експериментима, а то су појава различитих „фаза“: кинетичке, фазе смицања и фазе кластеровања. Упоредили смо зависност нормиране кинетичке температуре од времена у новој временској скали резултата добијених симулацијама са оним добијеним теоријским разматрањима. У симулацијама су коришћене и симулације са чврстим зидовима и оне са периодичним граничним условима. Такође смо приказали уређеност брзина у различитим фазама помоћу хистограма.

Даља истраживања би могла да иду у више смерова. Ми смо већ поменули неке, који се већ на основу постојећих кодова могу лако генерализовати. То је увођење гравитације као и могућност израде симулације у три димензије. Постоје и напори да се ураде симулације која би укључивале и силу трења, односно ротацију честица, међутим тада се ситуација доста компликује због нетривијалног проблема решавања нееластичног колапса.

Што се тиче реалних експеримената, они захтевају посебну апаратуру и брзе камере којима би се пратио процес. Кинетичка фаза је и у реалним експериментима добро проучена, док се остале фазе због своје крајње случајне природе још увек примарно проучавају нумерички.

10 Литература

1. B. Andreotti, Y. Forterre, and O. Pouliquen, *Granular media : between fluid and solid*, Cambridge University Press, 2013
2. H. M. Jaeger, S. R. Nagel, and R. P. Behringer, *Granular solids liquids and gasses*, Rev. Mod. Phys., Vol. 68, No. 4, October, 1996
3. S. McNamara and W. R. Young, *Dynamics of a freely evolving, two-dimensional granular medium*, Phys. Rev. E 53, 1996
4. T. Pöschel and T. Schwager, *Computational Granular Dynamics*, Springer, 2005
5. S. McNamara and W. R. Young, *Inelastic collapse in two dimensions*, Phys. Rev. E 50, 1994
6. P. K. Haff, *Grain flow as a fluid-mechanical phenomenon*, J. Fluid Mech. **134**, 401, 1983
7. J. J. Brey, M. J. Ruiz-Montero, and D. Cubero, *Homogeneous cooling state of a low-density granular flow*, Phys. Rev. E 54, 1996
8. J. J. Brey, M. J. Ruiz-Montero, and F. Moreno, *Validity of the Boltzmann equation to describe low-density granular systems*, Phys. Rev. E 69, 2004
9. A. Mehta and S. F. Edwards, *Statistical mechanics of powder mixtures*, Physica A 157 (1989)
10. P. Résibois and M. D. Leener, *Classical Kinetic Theory of Fluids*, A Wiley-Interscience Publication, New York, 1977
11. J. P. Hansen and I. R. McDonald, *Theory of Simple Liquids*, Academic Press; 2nd edition, 1990
12. Monasson and Pouliquen, *Entropy of particle packing : an illustration on a toy model*, Physica A, Volume 236, Issues 3–4, 1997
13. Knight *et al.*, *Density relaxation in a vibrated granular material*, Phys. Rev. E 51, 1995
14. N. V. Brilliantov and T. Pöschel, *Kinetic Theory of Granular Gases*, Oxford University Press, 2003.
15. I. Goldhirsch and G. Zanetti, *Clustering Instability in Dissipative Gases*, Physical Review Letters 70, 1993