

استخدام برنامج MATLAB لإيجاد حل تقريبي لمعادلة بولتزمان الحركية لسائل الكتروني

الدكتور نجاح قبلان*
الدكتور محمود أحمد**
عبد الناصر نعلاني***

(تاريخ الإيداع 29 / 6 / 2010. قُبِلَ للنشر في 11 / 10 / 2010)

□ ملخص □

تم في هذا البحث حل معادلة بولتزمان الحركية عددياً باستخدام برنامج MATLAB لإيجاد تحولات تابع توزع فرمي ديراك والكثافة الاحتمالية بتابعة كل من درجة الحرارة والسرعة والطاقة الحركية . كما حصلنا على المنحنيات الموافقة لهذه التحولات في فراغ ذي ثلاثة أبعاد دون وضع أية قيود على عبارة الطاقة التي قد تحتوي على بارامترات لاندوا من أية مرتبة. ثم قمنا بمقارنة النتائج التي حصلنا عليها مع النتائج المرجعية التي سبق الحصول عليها بطريقة مونت - كارلو (MC) والتي تعتمد على توزع ماكسويل بولتزمان، فوجدنا أن نتائجنا أعم وأكثر دقة، ويمكن الحصول على النتائج المرجعية لطريقة (MC) بأجراء التقريب $T \rightarrow \infty$.

الكلمات المفتاحية: معادلة بولتزمان الحركية . تابع توزع فرمي ديراك . البرنامج MATLAB . طريقة مونت كارلو . الطرائق العددية لحل المعادلة الحركية.

* أستاذ مساعد . قسم الفيزياء . كلية العلوم . جامعة تشرين . اللاذقية . سورية.
** أستاذ مساعد . قسم الفيزياء . كلية العلوم . جامعة تشرين . اللاذقية . سورية.
*** قائم بالأعمال . قسم الفيزياء . كلية العلوم . جامعة تشرين . اللاذقية . سورية.

Use The Matlab Program to Find an Approximate Solution for Kinetic Boltzmann Equation of an Electronic Liquid

Dr. Najah Kaban^{*}

Dr. Mahmoud Ahmad^{**}

Abdul Nasser Nasani^{***}

(Received 29 / 6 / 2010. Accepted 11 / 10 / 2010)

□ ABSTRACT □

This program allows us to find the variation of distribution function (probability density) as a function of other variables, such as temperature, velocity and kinetic energy. Furthermore to draw the correspondence charts (curves) in space consist of three dimensions without any restrictions on energy which can contain Landau parameters of higher numbers. Comparing these results with the results of Monte-Carlo method (MC), we can conclude that our results are more general if compared with the results in literature, it means that the MATLAB program we used for this purpose is more powerful to solve such problems in comparison to other ones.

Key Words: Kinetic Boltzmann equation – Fermi Dirac distribution function – MATLAB program – Monte Carlo method – Numerical methods to solve the kinetic equation.

^{*}Associate Professor, Department of physics, Faculty of Science, Tishreen University, Lattakia, Syria.

^{**}Associate Professor, Department of physics, Faculty of Sciences, Tishreen University, Lattakia, Syria.

^{***}Academic Assistant, Department of physics, Faculty of Sciences, Tishreen University, Lattakia, Syria.

مقدمة:

وضع بولتزمان معادلته الحركية في العام (1872) لوصف الخواص الديناميكية للغازات وكانت هذه المعادلة الأساس الذي بنيت عليه النظرية الحركية لاحقاً، والتي تصف سلوك الغاز المثالي والغاز المتحلل بالإضافة للسوائل. استخدمت هذه المعادلة لتعيين المقادير الفيزيائية للأجسام الصلبة مثل: الطواعية المغناطيسية، والمقاومة الكهربائية، والانضغاطية، ومعامل الانتشار في المواد وطاقة موجة السبين، كما تستخدم في دراسة علاقات التشنت وتخامد لاندواو في البلازما --- الخ [1-4] .

طور لاندواو ومن بعده سيلين هذه المعادلة لتصف الخواص الفيزيائية للجمل شبه الكلاسيكية، واستخرج منها ظاهرة أمواج السبين البارامغناطيسية والصوت الصفري والأول.... الخ [5-7] ، ثم طورت من قبل أبريكوسوف وتشيرفونكو لحساب معاملات الانتشار في المعادن [8,9] وفي تحديد الصيغ المختلفة لانتشار الأمواج المغناطيسية في المعادن القلوية [10,11] .

وإذا تتبعنا البحوث المنشورة حتى الآن والتي تعتمد طريقة المعادلة الحركية في تفسير الظواهر المجهرية، نجد أنها تشكل إضافة لطريقة تابع غرين، إحدى أهم الطرائق في دراسة وفهم الظواهر الطبيعية. عمّت المعادلة الحركية استناداً إلى فرضيات لاندواو في سائل فرمي من أجل مصفوفة الكثافة المغناطيسية $M(\underline{k}, w)$ و مصفوفة الكثافة $\rho(\underline{k}, w)$ بدون تقريبات، وتم فصل هذه المعادلة إلى جزأين يصف أحدهما تطور الكثافة المغناطيسية في الفراغ الطوري، كما يصف الآخر تطور الكثافة الجسيمية في نفس الفراغ [12] .

تكوّن هذه المعادلة بصيغتها العامة معادلة تفاضلية جزئية من الشكل:

$$\frac{\partial \varphi}{\partial t} + \nabla \cdot \vec{F}_D + \nabla \cdot \vec{F}_C + \nabla \Phi = 0$$

حيث:

Φ ، φ : عبارة عن مصفوفة مؤلفة من عمود لجملية محافظة مستقرة في وضع التوازن .
 \vec{F}_D, \vec{F}_C : عبارة عن تنسوران لهما نفس العدد من الأسطر، و يوافق عدد الأعمدة عدد درجات الحرية لهذه الجملية. أما الدليلين السفليين D و C فيقصد بهما المشتق ودوار التنسوران الموافقان، ولمزيد من المعلومات يمكن العودة إلى [13]. أما الحلول التحليلية المتوفرة لهذه المعادلة فتتم وفق تقريبات زمن التراخي Relaxation Time Approximation وبعدد محدود من الحدود لايتجاوز الثلاثة حتى الآن . علماً أنه لا يمكن حل هذه المعادلة تحليلياً بشكل كامل ومع تقدم الطرائق الحاسوبية والبرامج التي تساعد على حل المعادلات الرياضية في الفيزياء، فقد أمكن حل معادلة بولتزمان الحركية في العديد من الحالات اعتماداً على طريقة مونت كارلو [14,15] .

ويظهر البرنامج التقني MATLAB الذي يحتوي في داخله على برامج فرعية وأوامر يمكن استخدامها في حل المعادلات التفاضلية الجزئية، فإننا سنستخدم هذا البرنامج في تنفيذ حل المعادلة الحركية، ومن ثم مقارنة نتائجنا مع تلك التي تم الحصول عليها باستخدام طريقة مونت كارلو [16-18] .

أهمية البحث وأهدافه:

يهدف هذا البحث إلى استخدام الطرائق العددية لحل المعادلة الحركية بواسطة الحاسوب خطوة خطوة انطلاقاً من ثلاثة حدود ثم أربعة وفق شروط ابتدائية ملائمة. تعطى المعادلة الحركية بالشكل العام التالي:

$$\frac{\partial f}{\partial t} + v \cdot \nabla_r f + a \cdot \nabla_v f = \Omega \quad (1)$$

حيث: $f(r, v, t)$: تابع توزع الجسيم، r : موضع الجسيم، v : سرعة الجسيم، a : تسارع الجسيم، t : الزمن، $\Omega(r, v, t)$: تابع يدل على تكامل التصادم بين الجسيمات. وبأخذ $\Omega(r, v, t) = 0$ ، $a = 0$ نحصل على الشكل

$$\frac{\partial f}{\partial t} + v \cdot \nabla_r f = 0 \quad (2)$$

قمنا في البداية بحل المعادلة المختزلة (2) وذلك بكتابة برنامج (شيفرة برمجية code) يتم تنفيذه باستخدام البرنامج التقني MATLAB. نأخذ تابع توزع فرمي-ديراك التالي:

$$f(\varepsilon) \equiv u(T, \varepsilon) = \frac{1}{1 + e^{(\varepsilon - \mu)/k_B T}} \quad (3)$$

حيث يتم استبدال رمز تابع التوزيع فرمي-ديراك $f(\varepsilon)$ إلى الرمز $u(T, \varepsilon)$ بما يلائم نموذج المعادلة التفاضلية الجزئية وفقاً لما هو معرف في MATLAB.

$$\varepsilon = \mu \Rightarrow u(T, \mu) = u_0 = 0.5 \quad \text{الشروط الابتدائية:}$$

الشروط الحدية:

$$T = T_f \Rightarrow \frac{\partial u(T_f, \varepsilon)}{\partial T} = \frac{(\varepsilon - \mu)}{k_B T_f^2} \cdot \frac{e^{(\varepsilon - \mu)/k_B T_f}}{\left[1 + e^{(\varepsilon - \mu)/k_B T_f}\right]^2}, \quad T = T_0 \Rightarrow u(T_0, \varepsilon) = 0$$

يسمح لنا هذا البرنامج بإيجاد تحولات تابع التوزيع (الكثافة الاحتمالية) بدلالة المتحولات الأخرى مثل: درجة الحرارة والسرعة والطاقة الحركية، وبالتالي رسم المنحنيات المقابلة في فراغ ذي ثلاثة أبعاد.

لذلك يمكن لهذا البحث أن يكون قاعدة للانطلاق مستقبلاً لحل المعادلات الرياضية في الفيزياء بشكل تام وبدون تقريبات وإعطاء قيم عددية للحلول التحليلية المعروفة والتي يمكن استخدامها في التطبيق المباشر. يمثل هذا البحث الخطوة الأولى في الطريق إلى الاستخدام العملي لبرنامج MATLAB لتحديد وحساب المقادير الفيزيائية عن طريق حل المعادلة الحركية عددياً ورسم تحولات هذه المقادير فراغياً، وهذا ما يتيح لنا البرنامج التقني MATLAB

طرائق البحث ومواده:

استخدمنا في هذا البحث الطرائق البرمجية لحل معادلة بولتزمان الحركية، والتي تبني من خلال كتابة البرنامج الرئيسي. تم استخدام أداة الحل Solver:pdepe التي يمتلكها برنامج MATLAB بإدخال المعطيات اللازمة والتعويض عن الوسطاء بقيم معلومة لحل المعادلات التفاضلية الجزئية PDE والتي تقوم بدورها باستدعاء برامج فرعية

من داخل البرنامج الرئيسي لحل تلك المعادلة التفاضلية الجزئية. بعد الحصول على الحلول العددية لهذه المعادلة قمنا برسمها بدلالة متحولات المسألة المطروحة من قبلنا.

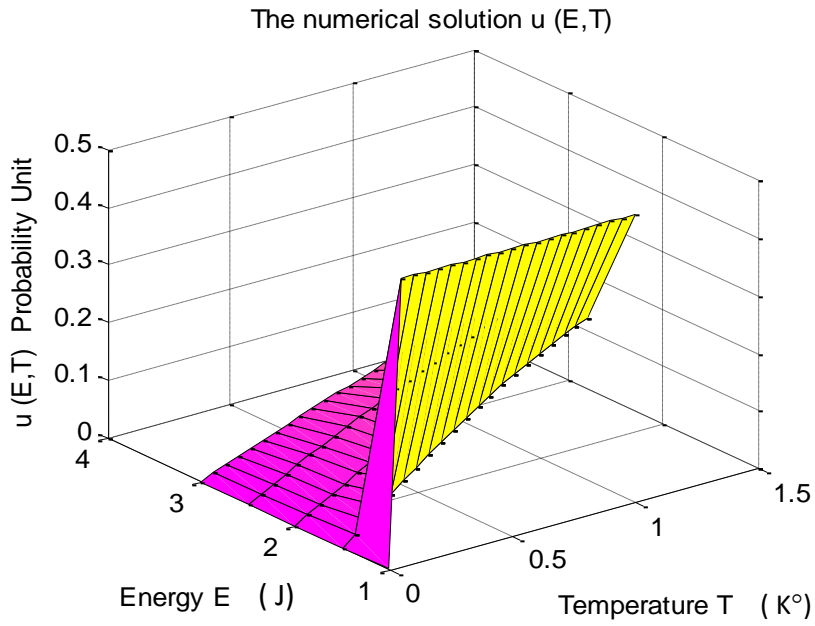
نورد الآن الشيفرة البرمجية المكتوبة ببرنامج MATLAB والتي تقوم بحل المعادلة (2) وقد استخدم فيها

الأوامر التالية :

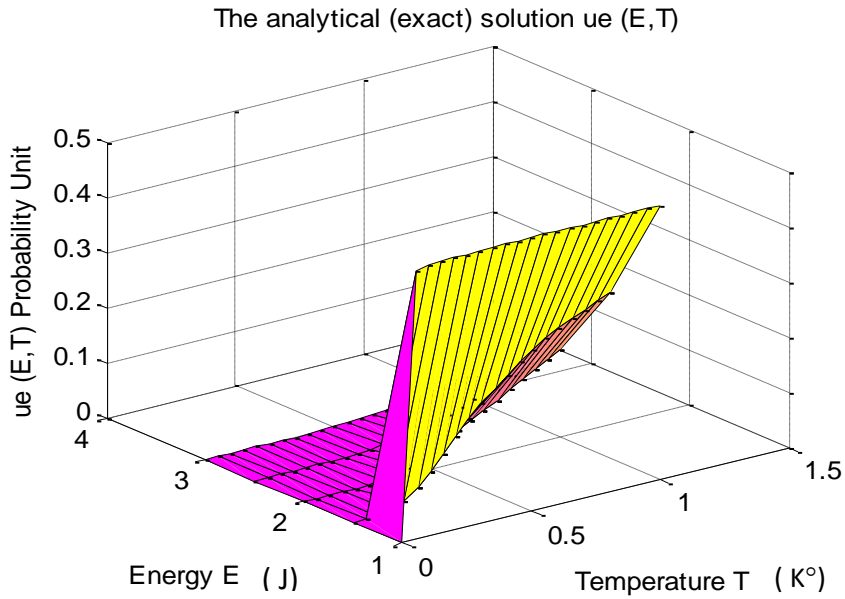
Meshgrid , figure , plot , grid on , surf , label , title .

الشيفرة البرمجية (1) :

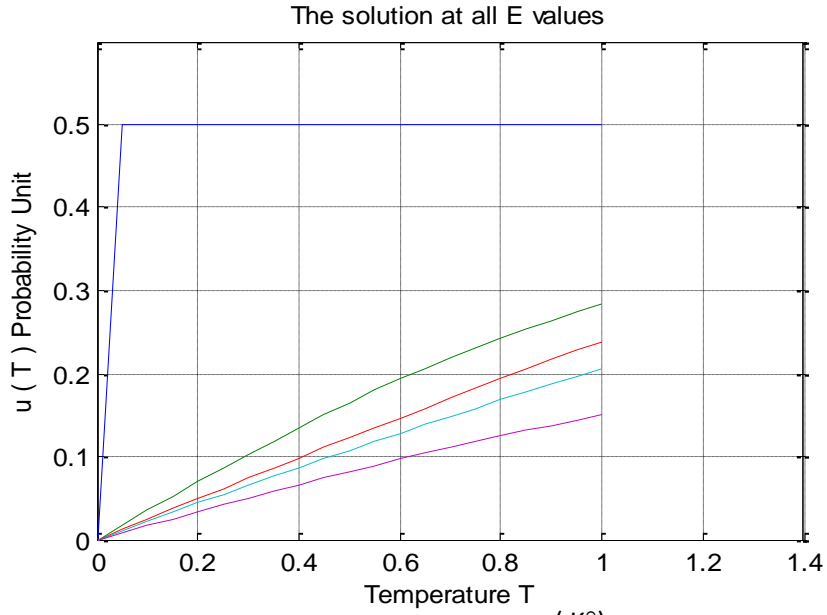
```
function pdex61
m = 0;mu=1;KB=0.6;T0=0.0001;Tf=1;E0=0.001;
T = T0:0.05:1+T0; E=E0+mu:0.5:3+E0
sol = pdepe(m,@pdex61pde,@pdex61ic,@pdex61bc,T,E,[ ],mu,KB,T0,Tf);
u = sol(:,:,);
figure;surf(T,E,u); title('Numerical solution');
xlabel('Temperature T');ylabel('Energy E');zlabel('Solution u(E,T)');
figure;plot(T,u(:,:,));grid on;
title('Solution at all E values');
xlabel('Temperature T');ylabel('u(T)');
figure;plot(E,u(:,:,));grid on;
title('Solution at all T values');
xlabel('Energy E');ylabel('Solution u(E)');
% -----
[T,E] = meshgrid(T,E);
ue = 1./(exp((E-mu)./(KB.*T))+1);
figure;surf(T,E,ue);
title('Analytical(Exact)solution');
xlabel('Temperature T');ylabel('Energy E');zlabel('Solution ue(E,T)');
% -----
function [c,f,s] = pdex61pde(T,E,u,DuDT,mu,KB,T0,Tf)
c = 1;
f=DuDT;
s =0;
% -----
function u0 = pdex61ic(T,mu,KB,T0,Tf)
u0 = 0.5;
% -----
function [plf,qlf,pr,qr] = pdex61bc(Tlf,ulf,Tr,ur,E,mu,KB,T0,Tf)
plf = ulf;
qlf = 0;
w= exp((E-mu)./(KB.*Tf));
pr = (E-mu)./(KB.*Tf.^2).*w./(w+1).^2;
qr = -1;
```

المخططات البيانية:

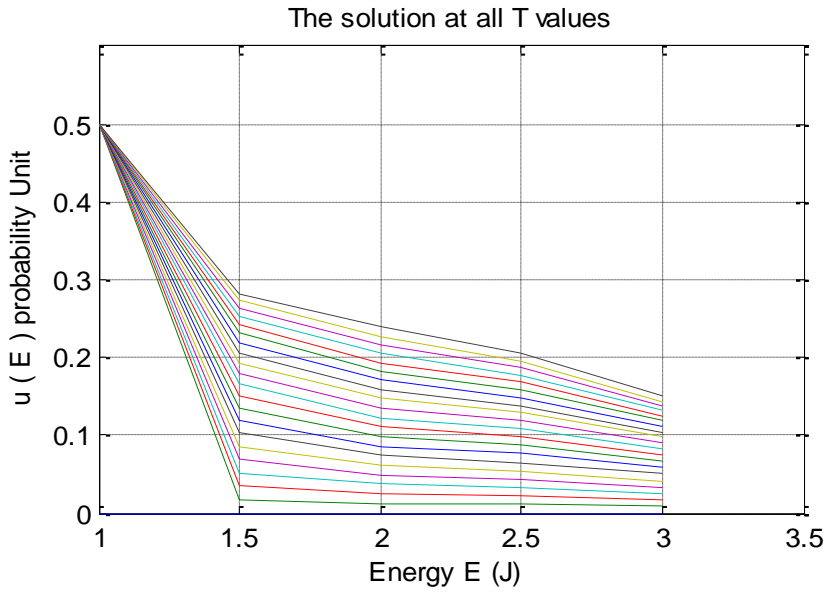
الشكل (1) . رسم فراغي لتابع التوزيع u بدلالة درجة الحرارة T والطاقة E والذي يمثل الحل العددي للمعادلة (2)



الشكل (2) . رسم فراغي لتابع توزع فرمي - ديراك u_e بدلالة درجة الحرارة T والطاقة E وفق المعادلة (3) والذي يمثل الحل التحليلي للمعادلة (2)



الشكل (3) . رسم جاذ : u بدلالة درجة الحرارة T ومن أجل عدة قيم للوسيط E (K°)



الشكل (4) . رسم جانبي للحل العددي u بدلالة الطاقة E ومن أجل عدة قيم للوسيط T

في الخطوة التالية من البحث تم إدخال حركة الجسيمات وذلك بأخذ: $\Omega(r, v, t) = 0$, $a \neq 0$ من المعادلة العامة (1) ، حيث تم الحصول على المعادلة التالية:

$$\frac{\partial f}{\partial t} + v \cdot \nabla_r f + a \cdot \nabla_v f = 0 \quad (4)$$

قمنا أيضاً بحل هذه المعادلة بعد إضافة الحد الثالث إلى الطرف الأيمن من المعادلة (4) في البرنامج (الشيفرة البرمجية code) السابق.

نعرض الآن الشيفرة البرمجية بشكلها الجديد والتي احتوت على أمر إضافي هو: diff إضافة للأوامر الآتفة الذكر.

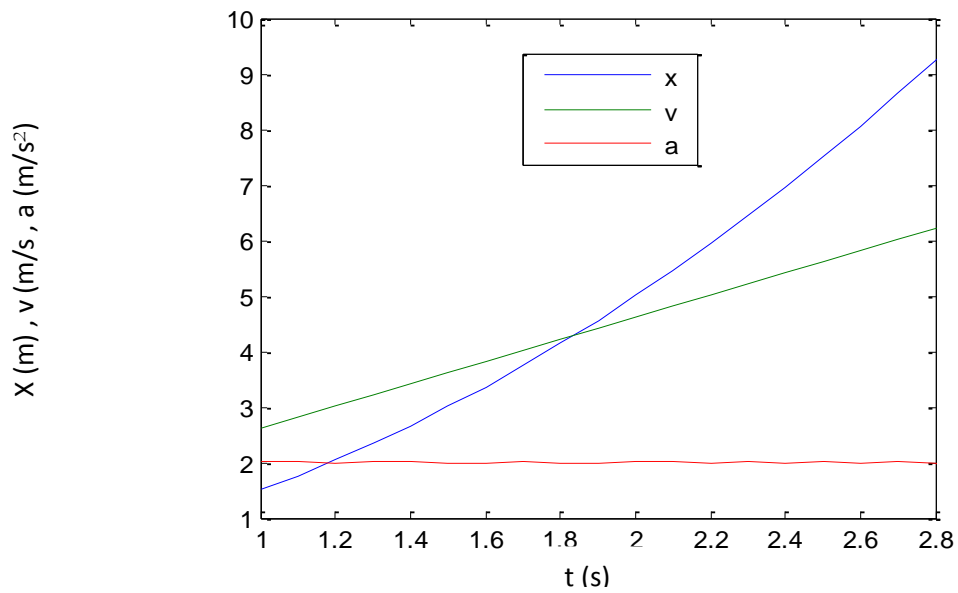
الشفرة البرمجية (2) :

```

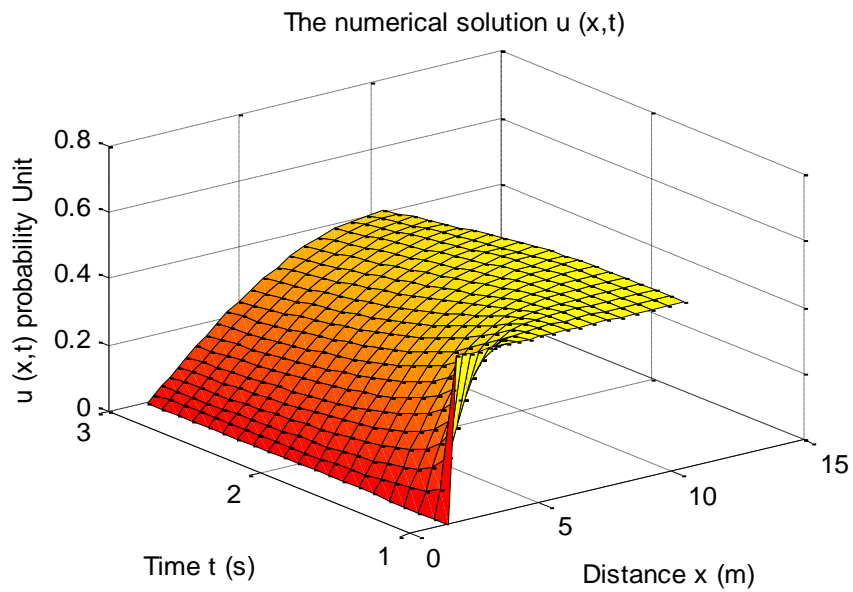
function pdex612
m = 0;mu=1;KB=0.6;x0=0.0001;t0=0.001;tf=3;n=20;
h=(tf-(t0+mu))/n;
t = t0+mu:h:tf+t0;
x = t.^2+1/2*t+x0;
xf=tf.^2+1/2*tf+x0;
v=diff(x)./diff(t);
v(:,n+1)=v(:,n)+h;
a=diff(v)./diff(t);
a(:,n+1)=a(:,n)+h;
figure;plot(t,x,t,v,t,a);
% -----
sol = pdepe(m,@pdex612pde,@pdex612ic,@pdex612bc,x,t,[ ],mu,KB,x0,xf,v,a);
u = sol(:,:,);
figure;surf(x,t,u);
title('Numerical solution');
xlabel('Distance x');ylabel('Time t');zlabel('Solution u(x,t)');
figure;plot(x,u(:,:));grid on;
title('Solution at all t values');
xlabel('Distance x');ylabel('u(x)');
figure;plot(t,u(:,:));grid on;
title('Solution at all x values');
xlabel('Time t');ylabel('Solution u(t)');
% -----
[x,t] = meshgrid(x,t);
ue = 1./(exp((t-mu)./(KB.*x))+1);
figure;surf(x,t,ue);
title('Analytical(Exact)solution');
xlabel('Distance x');ylabel('Time t');zlabel('Solution ue(x,t)');
% -----
function [c,f,s] = pdex612pde(x,t,u,DuDx,mu,KB,x0,xf,v,a)
c = 1;
f1=DuDx;
f2=v*f1;
f3=f1*(v/a);
f4=a*f3;
f=f2+f4;
s =0;
% -----
function u0 = pdex612ic(x,mu,KB,x0,xf,v,a)
u0 = 0.5;
% -----
function [plf,qlf,pr,qr] = pdex612bc(xlf,ulf,xr,ur,t,mu,KB,x0,xf,v,a)
plf = ulf;
qlf = 0;
w= exp((t-mu)./(KB.*xf));
pr = (t-mu)./(KB.*xf.^2).*w./(w+1).^2;
qr = -1;

```

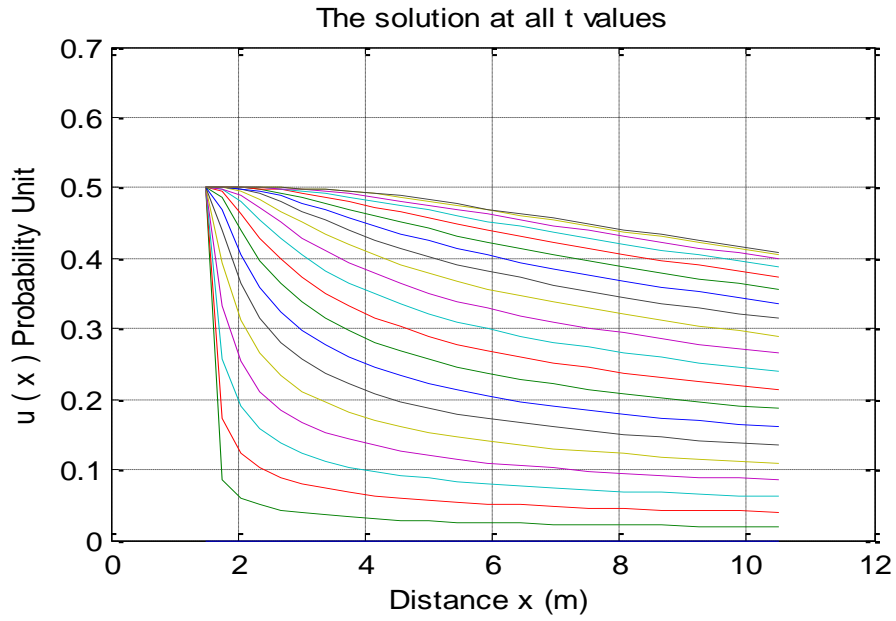

الحلول ممثلة بالمخططات البيانية:



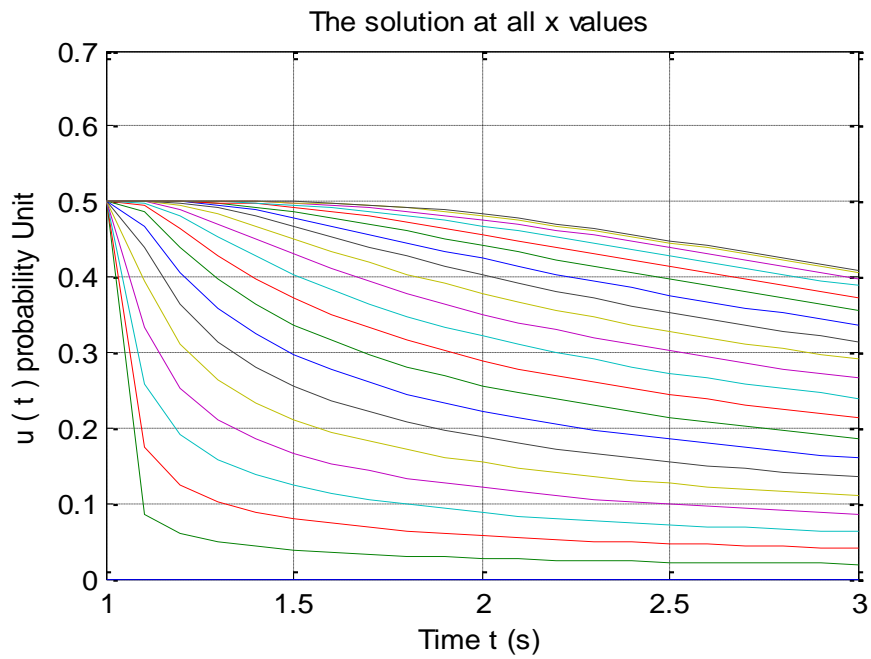
الشكل (5) . رسم بياني لتطور كل من الموضع x والسرعة v والتسارع a للجسيم بدلالة الزمن t



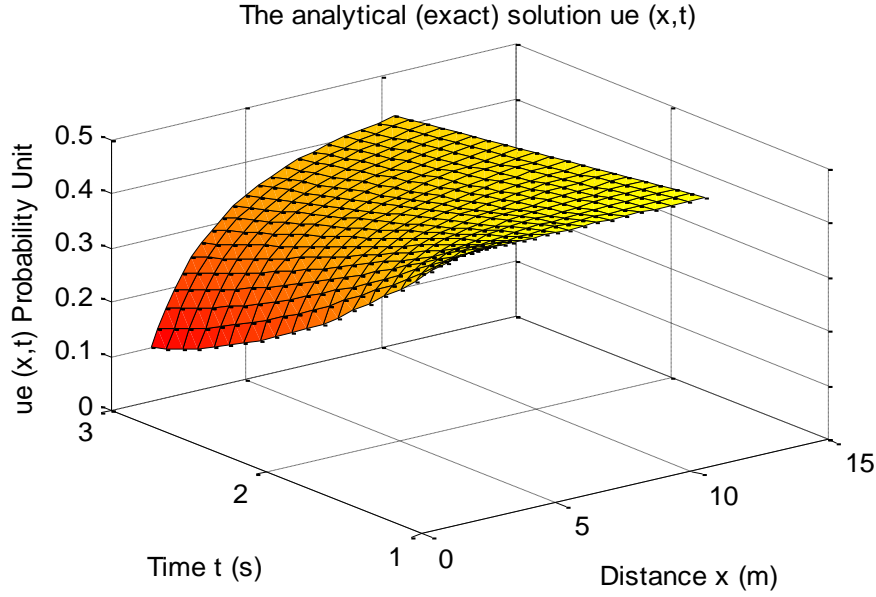
الشكل (6) . رسم فراغي لتابع التوزيع u بدلالة الموضع x والزمن t والذي يمثل الحل العددي للمعادلة (4)



الشكل (7) . رسم جانبي للحل العددي u بدلالة موضع الجسم x ومن أجل عدة قيم للوسيط t



الشكل (8) . رسم جانبي للحل العددي u بدلالة الزمن t ومن أجل عدة قيم للوسيط x



الشكل (9) . رسم فراغي لتابع توزع فرمي . ديراك u_e بدلالة موضع الجسم x والزمن t وفق المعادلة (3) بعد التغير الذي طرأ عليه عند إدخال حركة الجسيمات فيه

الاستنتاجات والتوصيات:

الاستنتاجات:

لازال حل معادلة بولتزمان الحركية أو معادلة لانداو . سيلين في سائل فرمي في بداياته سواء أكان الحل تحليلياً أم عددياً، ولم تحل هذه المعادلة في كلتا الحالتين إلا جزئياً.

استخدمت طريقة مونت كارلو لحل هذه المعادلة عددياً من أجل عدة تقريبات مثل تقريب زمن التراخي واختصار عدد الحدود الداخلة في هذه المعادلة، فقد تم حلها بثلاثة حدود فقط حتى الآن [19-24] . ونظراً لطبيعة هذه الطريقة العشوائية أي اعتمادها على أعداد وتوزيعات عشوائية، فإن الدقة في حل الجمل المعقدة بهذه الطريقة يزيد الخطأ المرتكب كثيراً. ومع تقدم البرامج الحاسوبية وبصورة خاصة البرنامج التقني MATLAB الذي صمم خصيصاً ليبي متطلبات الفيزيائيين والمهندسين حيث يوصف المقدار الفيزيائي فيه بواسطة مصفوفة كيفية الأبعاد، ويتمتع بمجموعة كبيرة من البرامج الفرعية Subroutines وأدوات الحل Solvers خاصة أداة حل المعادلات التفاضلية الجزئية Solver:pdepe والتي تم استخدامها هنا. ونظراً للميزات الكثيرة والهامة التي يتمتع بها هذا البرنامج قمنا باستخدامه لحل معادلة بولتزمان الحركية وهو ما يمكن تطويره بصورة مستمرة لاحقاً مع التطور المستمر لهذا البرنامج .

بالعودة إلى المخططات (3,4,7,8) ، والتي تمثل تطور الكثافة بدلالة متحولات تابع التوزع أي درجة الحرارة والطاقة، وإجراء التقريب $2D \rightarrow 1D$ يتضح التوافق التام مع منحنيات فرمي ديراك وفق بعد واحد عند درجات حرارة منخفضة، كما أنها تتطابق مع ما توصلت إليه طريقة مونت كارلو في أحسن حالاتها التي عولجت من خلالها على الرغم من أننا لم نستخدم كل إمكانيات البرنامج حتى الآن، وهذا ما نأمل بإتمامه في دراسة لاحقة. لقد تمكنا في البرنامج الذي بنينا هنا من ملائمة معادلة بولتزمان مع أداة الحل في هذا البرنامج، واستطعنا أن ننفذ البرنامج من أجل تابع توزع فرمي . ديراك بدلاً من تابع توزع مكسويل . بولتزمان الذي استخدمه آخرون بطريقة مونت كارلو .

يمكن تنفيذ هذا البرنامج من أجل أية معادلة تفاضلية جزئية باستخدام تابع توزع فرمي . دبراك كمصفوفة كثافة ونستطيع رسم المخططات البيانية لهذه الحلول باستخدام أوامر مناسبة في برنامج MATLAB نفسه. كما يعرف كل متحول من المتحولات المدروسة على أنه شعاع مكون من n مركبة، ويمكن لبرنامجنا أن يعرف على كل متحول من هذه المتحولات نظراً لسعة برنامج MATLAB . بالإضافة لذلك استخدمنا الشروط الابتدائية والشروط الحدية بما يتلاءم ومتحولات تابع التوزيع (FD). وبالرغم من التحسينات التي تدخل يومياً على إمكانيات الحاسوب، فإن حل معادلة بولتزمان الحركية لا يزال تحدياً معاصراً للفيزياء التطبيقية وبشكل خاص لتحديد طرائق الحل بدون تشوهات.

التوصيات:

تطور استخدام الإنسان للحاسوب في حل المسائل التقنية في الآونة الأخيرة تطوراً كبيراً حيث يمكن بناء المسائل التقنية في الحاسوب أولاً ومن ثم اختبارها ونمذجتها ودراسة الحلول وتجريبها على الحاسوب قبل بدء تنفيذها على أرض الواقع .

هنالك العديد من المحاولات التي تهتم بحل المعادلات الرياضية في الفيزياء مثل المعادلة الحركية ومعادلة شرودينغر ومعادلة لابلاس وبواسون وغيرها، إلا أنها لاتزال في البداية، وما قمنا به هنا من حل لمعادلة بولتزمان الحركية بأربعة حدود عن طريق كتابة برنامج رئيسي يتعرف على أوامر MATLAB إلا البداية في هذا المجال حيث نتطلع إلى حل المعادلة الحركية بحدود جديدة مثل حد زيمان وحد راشبا للترابط السبيني المداري وغيرها من الحدود التي تم إدخالها مؤخراً أثناء تعميم هذه المعادلة.

المراجع:

- 1- PLATZMAN, P.M.; WOLLFF, P.A. *in supplement,13,solid state physics* New york Academic press chapters 2,6, 1973, 9-11.
- 2- RICE,T.M.*the effects of Electron-Electron Interaction on the properties of metal*, Phys. Rev., 175,1968, 858.
- 3- Ahmad,M,2D *spin wave*,phys stat.solid(b),160,K65,1990, 8.
- 4- ABRIKOSOV, A.A. *Methods of quantum field theory in statistical mechanic*, DOVER publication NY chapter, 4, 1989, 154.
- 5- LANDAU, L.D.; EXSP, ZH.; TEOR, FIZ. 30,1058(1956) JEPT 3,920 (1957) .English trans: Sov. phys.
- 6- SILIN,V.P.; ZH EXSP,T. F, 33,495(1957) JEPT 6,387(1958) .English trans: Sov. phys.
- 7- BAYM, G.; PETHICK, C. *concepts and applications Landau Fermi-Liquid theory*.WILLEY-VCH Verlag & Co. KaA. Wenheim, Germany ,2004.
- 8- ABRIKOSOV, A.A, *Theory of normal metals*,Academic press,chapter(2,3,13), 1972.
- 9- CZERWONKO, J. *spin wave*, Japa. J. Appl. Phys (suppl.),26.223,1987.
- 10- GLADYSZ, S.; AHMAD, M. in. *Theory of paramagnetic spin waves in simple metals*, Jour. of. Low Tempe. Phys. V83,N1-2, 1991, 1-13.
- 11- KABALAN, N.; AHMAD, M. *Regenerate particle and magnetic density equations from general Landau-Silin kinetic equation*; Tishreen university Journal Basic Science Series to be published ,2010.
- 12- PARESCHI, L.; TRAZZI, S. *Numerical solution of Boltzmann equation by TRMC methods*.university of Ferrara,Via Machiavelli 35,44100 Ferrara Italy, E-Mail: pareschi@dm.unife.it. university of Ferrara, Via Machiavelli 35,44100 Ferrara Italy, E-Mail:trazzi@dm.unife.it.

- 13- KARL, R. ANSGAR, J.I. *Matrix Compression for Spherical Harmonics expansion of the Boltzmann Transport Equation for semiconductors*, ASC Report N 10./1010 Vienna University of Technology, www.asc.tuwien.ac.at . 34, 2010.
- 14- PAUL, L.; VRIES, DE. *Afirst course in computational physics*, John Wily&sons,INC,1994.
- 15- PANG, TAO. *An introduction to computational physics*,Cambridge university press 1997.
- 16- JENSEN, M. H. *Computational physics*, University of oslo, fall 2006.
- 17- MATLAB, *The language of Technical Computing*,www.mathworks.com comp.soft.sys. version 6, /matlab, 2001, 904.
- 18- MAE 502, *Partial differential equations in engineering*,Course website, <http://www.public.asu.edu/~hhuang> 38/MAE502.html spring 2009.
- 19- NICOLAS,G, *at all ,Low-Variance Monte Carlo Solution of the Boltzmann transport Equation*, Campridge, MA 02139 Cambridge, 2009 ,17.
- 20- MORAN,W.; SHIYI,C.I. *on Applicability of Poisson-Boltzmann Equation for Micro-and Nanoscale Electroosmotic flows,communications in computational physics* Vol. 3, N.5, 20081087-1099VSA.
- 21- KREMS, M. *The Boltzmann transport equation:Theory and applications*, December 10, 2007.
- 22- LACHISH, U. *Boltzmann transport equation*, Urita @internet- zahav.net, 2002.
- 23- NOZIERES, P. *Theory of interaction Fermi system*,Academic press (NY) chapter (1) ,1964.
- 24- ABRIKOSOV, A.A. *Fundamental of the theory of metal*,North-Holland Amesterdam Part I paragraphs , 2,13, 1988 , 24,263,276.