

# 臺灣二〇〇六年國際科學展覽會

科 別：物理科

作 品 名 稱：黑暗的力量

得 獎 獎 項：佳作

學 校 / 作 者：國立臺灣師範大學附屬高級中學 鍾岳霖

## 作者簡介



姓名：鍾岳霖

生日：1989/3/16

專長：數學、電腦、人際 EQ、合群、領導與組織能力

如果說我在復興學會如何走路，我在附中就學會如何昂然邁進；如果說復興教我如何在翻滾紅塵中孤然挺立，附中教我的就是如何在時代洪流中和群眾和而不同。

在復興畢業之後，依照學測成績而進入了附中就讀，在自由開放的附中，我雖不習慣但是很快的就在這裡找到自己的天地。延續著國中的能力，在附中也能利用自己原有的優勢繼續奮鬥。另一方面，附中也開啓了我的性中開朗樂觀的一面，讓我有了一座一個領導者的風範，也有做被領導的風度。

國中的短短三年讓我意識到自己在樹禮方面的潛能以及相當不錯的理解力；而昇上高中科教班後，經歷培養、競爭、鍛鍊琢磨，我更感謝附中闢了一條新路給我，讓我跳脫文科的框框，進入一個無限寬廣而更適合自己的世界。

# 目 錄

摘要	1
壹、前言	2
一、研究動機	2
二、研究目的	2
貳、實驗設備與器材	2
參、研究過程和方法	2
一、研究原理	2
二、研究方法	3
肆、研究結果	6
伍、討論	16
陸、結論	17
柒、參考資料及其它	17
附件	18

# Force in the Dark

## 黑暗的力量

### 英文摘要(Abstract)

We study the magic power of dark energy and dark matter by using theoretical derivation and numerical simulations. We found that:

1. The dark energy will gain kinetic energy from the moving dark matter through gravitational interaction. Due to the law of energy conservation, the motion of the dark matter will slow down and satisfy

$$E_k(t) - E_{k0} \propto \rho_{DE}^{1.92} t,$$

where  $E_k(t)$  is the kinetic energy of the dark matter,  $E_{k0}$  is its initial kinetic energy,  $\rho_{DE}$  is the energy density of the dark energy, and  $t$  is the time.

2. The formation history and the structure of galaxies will be different due to the existence of dark energy. The more the dark energy, the earlier the formation of the galaxy core. In addition, the kinetic energy  $E_k(R)$  as a function of  $R$  will be different if the  $\rho_{DE}$  is different. Thus we can observationally measure the  $E_k(R)$  of galaxies, compare it with our results here, and then deduce the  $\rho_{DE}$  in our universe.

The results here can be applied to the observations in the near future.

### 中文摘要

我們藉由理論的推導，配合電腦模擬的手段，來探討宇宙中黑暗物質和黑暗能量的神祕力量。我們發現：

- 一、黑暗能量會透過重力交互作用而從運動中的黑暗物質獲得力學能，而且因力學能守恆，致使黑暗物質的速率減慢，滿足

$$E_k(t) - E_{k0} \propto \rho_{DE}^{1.92} t$$

其中  $E_k(t)$  為黑暗物質的動能， $E_{k0}$  為其初始動能， $\rho_{DE}$  為黑暗能量的密度， $t$  為時間。

- 二、星系的形成過程及結構，會因黑暗能量的存在而改變。黑暗能量越多時，星系的核心會越早形成。而且動能  $E_k(R)$  隨著至星系中心距離  $R$  的變化，會因  $\rho_{DE}$  的不同而不同，因此可以試圖量測宇宙中星系的  $E_k(R)$ ，然後和這裡的結果比對，即可推導出宇宙中的  $\rho_{DE}$ 。這些研究成果，將可直接應用在未來的觀測結果上。

## 壹、前言：

### 一、研究動機

在展望系列演講以及科學人雜誌中得知，近年來在宇宙學領域中，最引起大家興趣的是，占宇宙總能量七成的**黑暗能量**和兩成半的**黑暗物質**，即使是對現今的宇宙物理學家而言，它們都還一直帶著神秘的色彩。尤其是黑暗能量，更是一反課本中所學的傳統牛頓思想，而提供「萬有斥力」，著實引人瑕思。這些新知引起我強烈的興趣，想要自己來看看它們之間的「萬有引力」和「萬有斥力」，到底會讓整個物理空間變成什麼樣子。

### 二、研究目的

本研究的目的，是要利用課本所學，藉由理論的推導，配合電腦模擬的手段，來探討黑暗物質和黑暗能量的作用力及相互關係，並進而了解一個運動中的質點所受這些黑暗力量的影響，以及宇宙中星系形成過程中所受這些黑暗力量的影響。了解黑暗能量及黑暗物質的性質，是目前世界上宇宙學領域中最熱門的主題之一，所以我們的研究成果，將可以在短期的未來，直接應用在這些期待中的觀測結果上，以揭開這些黑暗力量的神祕面紗。

## 貳、實驗設備與器材

一、軟體：C 語言程式、Matlab 分析軟體

二、硬體：Sun Blade 2000 電腦工作站、Pentium 個人電腦

## 參、研究過程和方法

### 一 研究原理：

本研究主要是利用牛頓的反平方引力定理

$$F = \frac{GMm}{r^2} \quad (\text{公式 1})$$

來探討黑暗物質和黑暗能量之間的萬有引力及萬有斥力，其中  $F$  為力， $G$  為重力常數， $M$  及  $m$  為黑暗物質或黑暗能量的質量， $r$  為兩質點間的距離。但根據愛因斯坦方程式，得知在式子中也必須考慮壓力，也就是  $m = \rho V$  中的  $\rho$  必須用  $(\rho + 3p)$  來取代，也就是

$$F = \frac{GMV(\rho + 3p)}{r^2} \quad (\text{公式 2})$$

對黑暗物質而言  $p = 0$ ，所以（公式 1）仍試用。對黑暗能量而言， $p = -\rho$ ，所以（公式 2）變為

$$F = \frac{-2GMm}{r^2} \quad (\text{公式 3})$$

因此萬有引力變為萬有斥力！

另外和本研究有關的重要觀念有：黑暗能量是組成宇宙最主要的部分，也是科學上最大的謎；它是一種斥力，使宇宙得以加速膨脹，而一般物質的重力則會使膨脹減速。因此，本研究所利用的關鍵，就是黑暗能量的負壓力，以致原本的萬有引力變為萬有斥力，而讓整個宇宙中物質的運動及分佈改變。

## 二、研究方法：

我們的研究方法，基本上是依以下順序，來探討黑暗物質及黑暗能量的運動及動力學。必要時我們使用 C 語言來完成所需的電腦多體模擬。

1、黑暗質量和黑暗能量的密度相同時，彼此運動的情形：

- (1) 利用 C 語言中的函數“rand()”來隨機在一箱子中分佈黑暗物質和黑暗能量。
- (2) 將放置在箱子中的這些粒子其座標輸出至檔案中，初速度為 0。
- (3) 依照（公式 1）及（公式 3），依序求出黑暗物質和黑暗物質間、黑暗能量和黑暗物質間、以及黑暗能量和黑暗能量間的交互作用力。
- (4) 求出每一個粒子所受的合力，利用

$$\mathbf{F} = m\mathbf{a} \quad (\text{公式 4})$$

的公式求出此粒子的加速度  $\mathbf{a}$ ，再利用

$$\mathbf{S} = \mathbf{v}dt + \frac{1}{2}\mathbf{a}dt^2 \quad (\text{公式 5})$$

求出此粒子在  $dt$  後的位移  $\mathbf{S}$ ，最後利用

$$\mathbf{v}(t + dt) = \mathbf{v}(t) + \mathbf{a}dt \quad (\text{公式 6})$$

出此粒子在  $dt$  後的速度  $\mathbf{v}$ 。

- (5) 經由 fprintf 的指令輸出粒子經過運算後的座標  $\mathbf{x}$  和速度  $\mathbf{v}$ 。
- (6) 利用迴圈來反覆執步驟(3)至(5)。

(7) 經由 Matlab 處理所輸出之各個時間的資訊，以將粒子的運動做成動畫，使我們更能清楚了解到在宇宙中這些物質或能量的動向和最後所形成的圖形。

2、將黑暗質量和黑暗能量的密度改爲宇宙中所觀測到的比 3：7，然後比照以上的 1 來模擬彼此運動的情形。

3、在眾多黑暗能量中放入一個黑暗物質的質點，觀察它的運動：

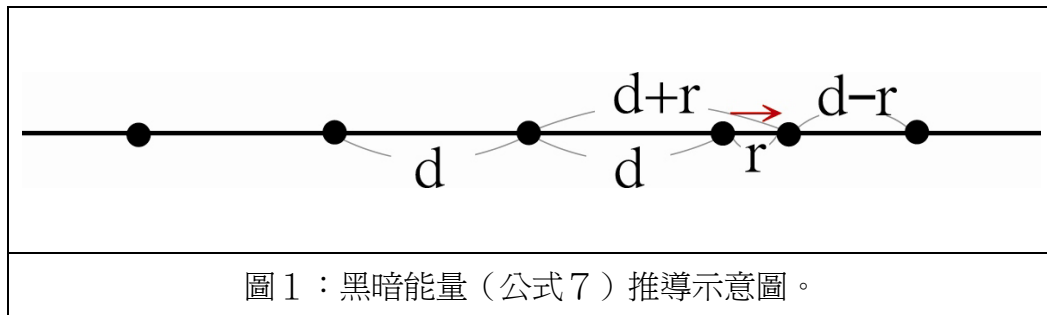
(1) 利用研究方法 1 的方式，計算黑暗物質和黑暗能量、黑暗能量和黑暗能量兩兩之間的作用力。

(2) 經由 Matlab 處理所輸出之各個時間的資訊，以將粒子的運動做成動畫，使我們更能清楚了解到在宇宙中這些物質或能量的動向和最後所形成的圖形。

4、在眾多黑暗能量中放入一個黑暗物質的質點，觀察它的運動（改變初速度）：

(1) 在程式中將黑暗物質的部分改爲只有一個粒子，並給這個粒子對某一軸有一初速度，而黑暗能量則全放置在箱子中的均勻格點上。

(2) 在黑暗能量互相排斥的部分上，利用我所計算出的公式將加速度算出。其計算過程如下：



利用（公式 3）及（公式 4），得知  $a = \frac{-2GM}{r^2}$ 。當某一個黑暗能量點相對於其原本的格點有一位移  $r$ （如上圖 1），則此質點所受到其左邊的各個質點之加速度和爲（方向爲向右）

$$\frac{2GM}{(r+d)^2} + \frac{2GM}{(r+2d)^2} + \dots + \frac{2GM}{(r+nd)^2} + \dots = 2GM \sum_{n=1}^{\infty} \frac{1}{n^2 d^2 \left(\frac{r}{nd} + 1\right)^2}$$

同理其所受到其右邊的各個質點之加速度和爲（方向爲向左）

$$\frac{2GM}{(r-d)^2} + \frac{2GM}{(r-2d)^2} + \dots + \frac{2GM}{(r-nd)^2} + \dots = 2GM \sum_{n=1}^{\infty} \frac{1}{n^2 d^2 \left(\frac{r}{nd} - 1\right)^2}$$

兩式相減之後得加速度為

$$a \cong \frac{8rGM}{d^3} \sum_{n=1}^{\infty} \frac{1}{n^3} = \frac{8rGM}{d^3} * Zeta(3) \cong \frac{rGM}{d^3} * 8 * 1.20206 \quad (\text{公式 7})$$

利用以上之（公式 7），將可大幅減少程式中的計算量。

- (3) 利用（公式 3）計算出黑暗物質質點的加速度之和，利用（公式 7）計算出黑暗能量質點間的作用力，然後利用（公式 5）及（公式 6）來算出它  $dt$  後的位移及速度。
- (4) 將黑暗物質及黑暗能量的座標、速度和時間輸出至檔案。
- (5) 利用迴圈來反覆執步驟(3)及(4)。
- (6) 改變黑暗物質的初速度  $v_0$ ，並重新執行步驟(1)至(5)，以探討初速是否是此運動行為中的關鍵。將運算結果分別輸出至別的檔案。
- (7) 利用 Matlab 將所有不同初速度黑暗物質的數據進行分析，以了解改變初速度後黑暗物質與黑暗能量之關係所受到的影響。

5、在眾多黑暗能量中放入一個黑暗物質的質點，觀察它的運動（改變黑暗能量的密度）：

- (1) 重覆執行以上 4 的步驟(3)至(5)，使用相同初速，但改變黑暗能量的密度。
- (2) 利用 Matlab 將所有不同初速度黑暗物質的數據進行分析，以了解改變黑暗能量密度時黑暗物質與黑暗能量之關係所受到的影響。

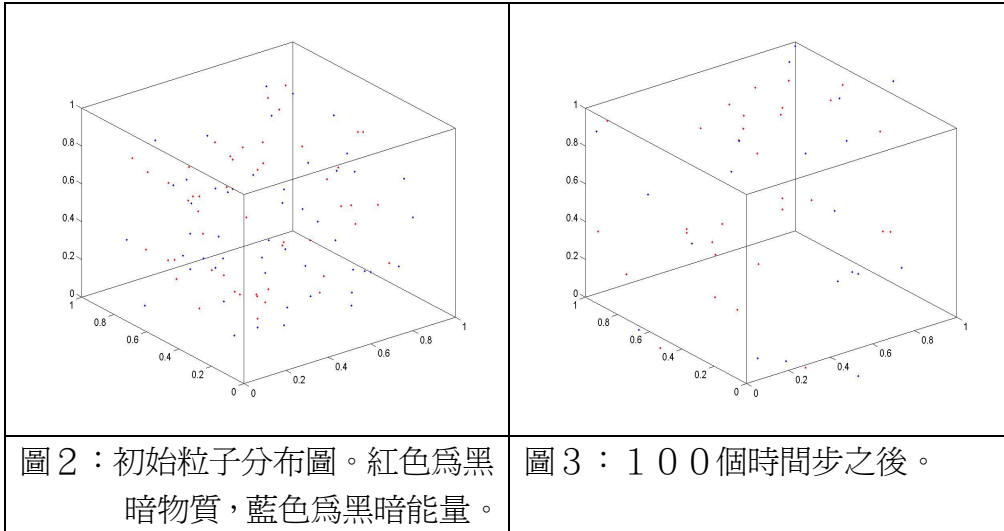
6、放置大量的黑暗物質，模擬宇宙中星系的形成。此部分分為兩部分，一個為具有黑暗能量，另一個為沒有黑暗能量：

- (1) 有黑暗能量：
  - A. 將黑暗能量皆放置於箱子中的格點上，其方法是利用三個迴圈來設定其黑暗能量的座標，並且利用畢氏定理來將 1000 顆黑暗物質放至於一中心座標為(0.5,0.5,0.5)，半徑為 0.2 的圓球範圍中。
  - B. 利用（公式 1）算出黑暗物質間的作用力，利用（公式 3）算出黑暗物質和黑暗能量間的作用力，利用（公式 7）算出黑暗能量間的作用力。
  - C. 求出各質點所受的和力，再利用（公式 5）及（公式 6）來算出它們在  $dt$  後的位移及速度。
  - D. 將黑暗物質及能量的位移、速度和時間輸出至檔案。
  - E. 利用迴圈來反覆執步驟 B 至 D。
  - F. 利用 Matlab 分析結果。
- (2) 沒有黑暗能量：
  - A. 比照 1 中的作法，但移除所有的黑暗能量，執步驟 A 至 E。
  - B. 利用 Matlab 分析結果。
- (3) 利用 Matlab 比較分析(1)及(2)之結果的異同。

## 肆、研究結果

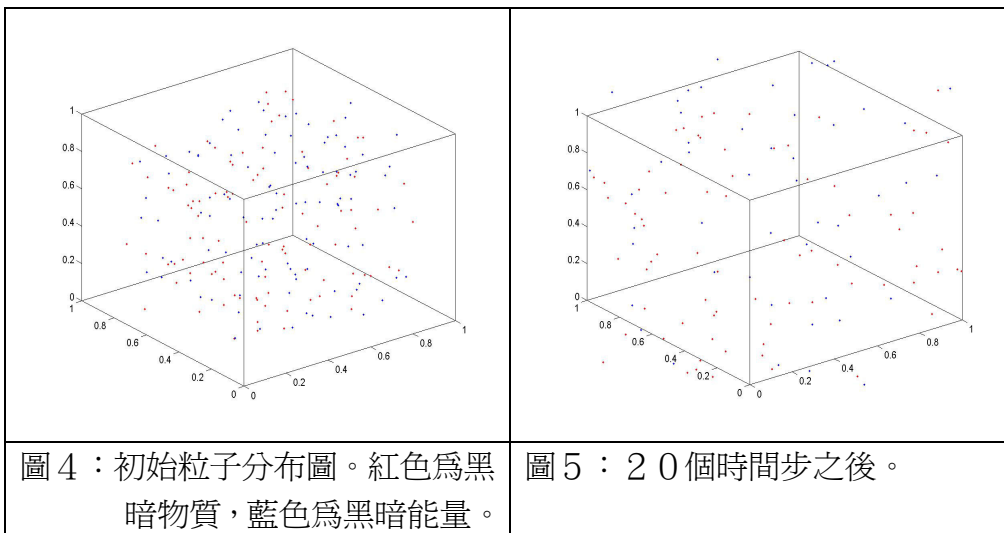
一、 黑暗物質和黑暗能量的密度相同：

以下圖 2 為初始的粒子分布圖，圖 3 為 1 0 0 個時間步（time step）之後的粒子分佈圖。



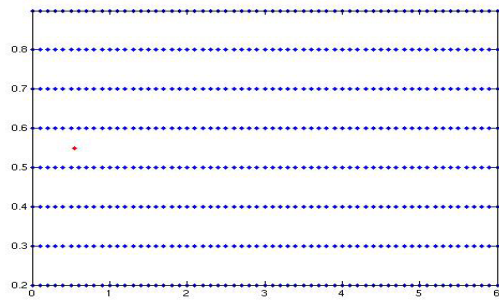
二、 黑暗質量和黑暗能量的密度改為宇宙中所觀測到的比 3：7 時：

以下圖 4 為初始的粒子分布圖，圖 5 為 2 0 個時間步（time step）之後的粒子分佈圖。

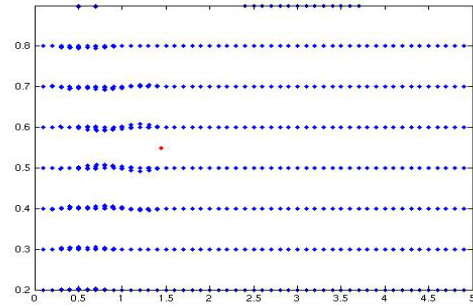


由以上一及二的結果中可看出，所有的粒子，包括黑暗物質及黑暗能量，在時間越長時都會向箱子的外部飛出，這表示萬有引力敵不過萬有斥力，以致所有粒子的淨力都像是斥力一般。其實這個結果，可由（公式 1）及（公式 3）之絕對值差兩倍的事實中可看出。

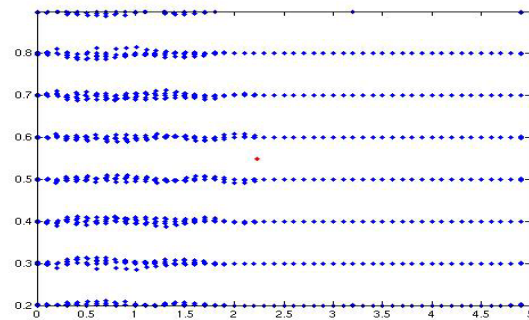
三、在眾多黑暗能量中放入一個黑暗物質的質點，觀察它的運動：



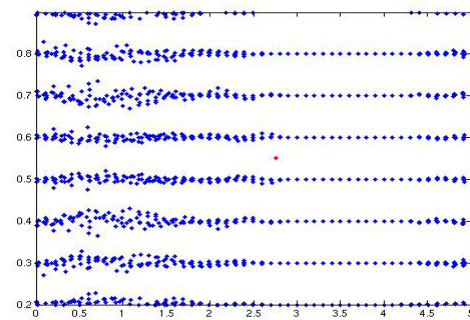
第 0 個時間步



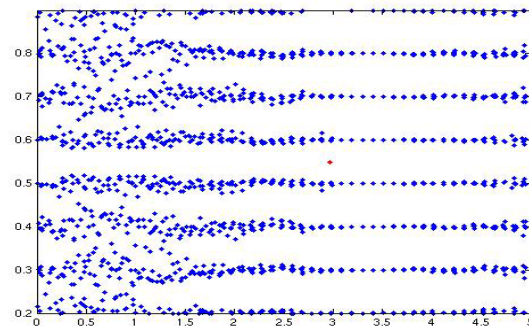
第 20 個時間步



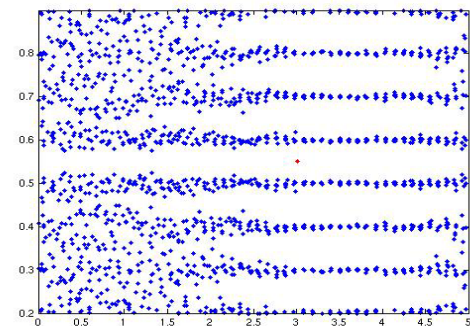
第 40 個時間步



第 60 個時間步



第 80 個時間步



第 100 個時間步

圖 6：橫軸及縱橫分別為質點的 x 及 y 座標。

我們可以進一步分析黑暗物質質點動能隨時間的變化，如圖 7 所示，橫軸為時間，縱軸為黑暗物質質點的動能。

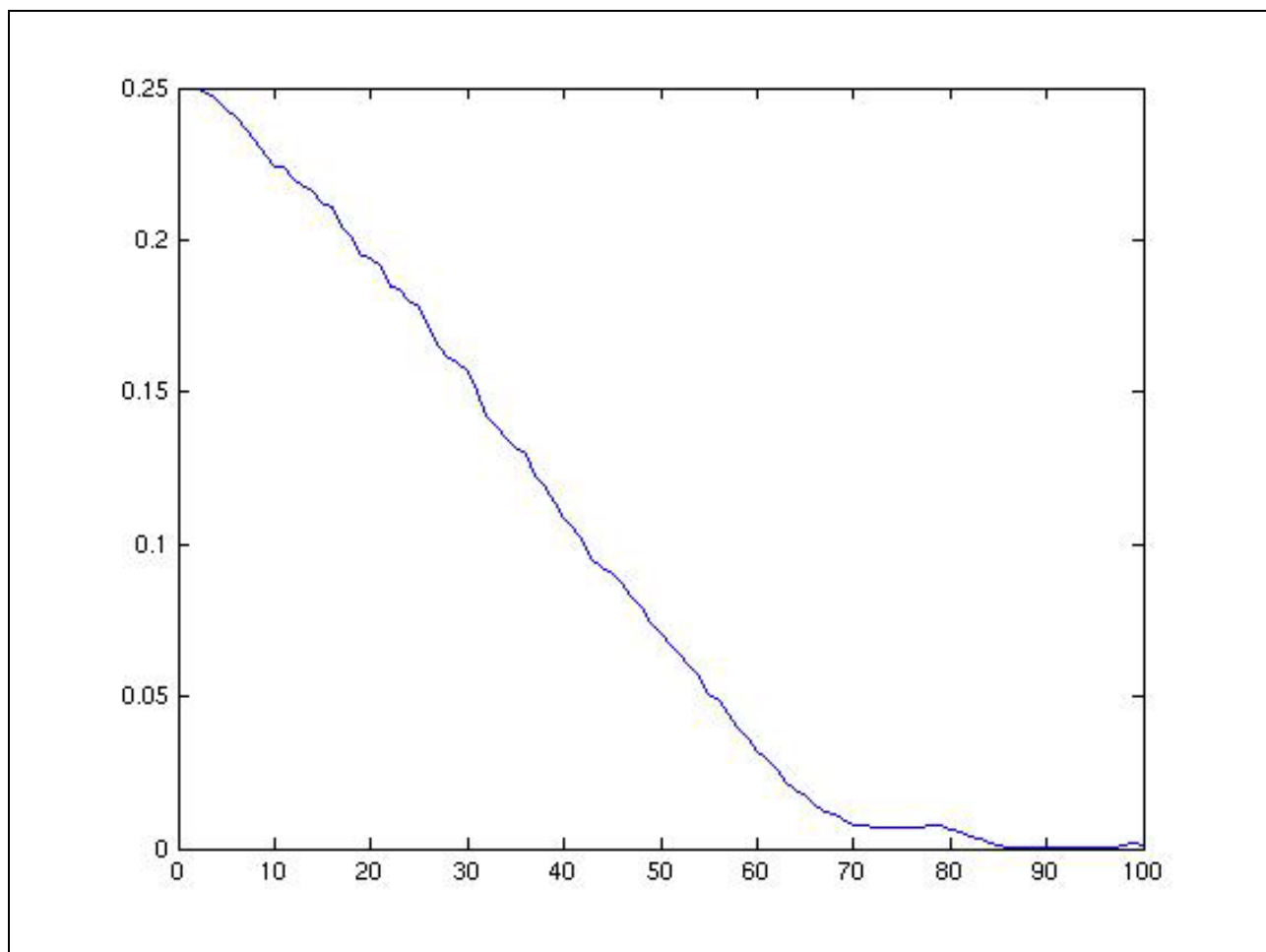
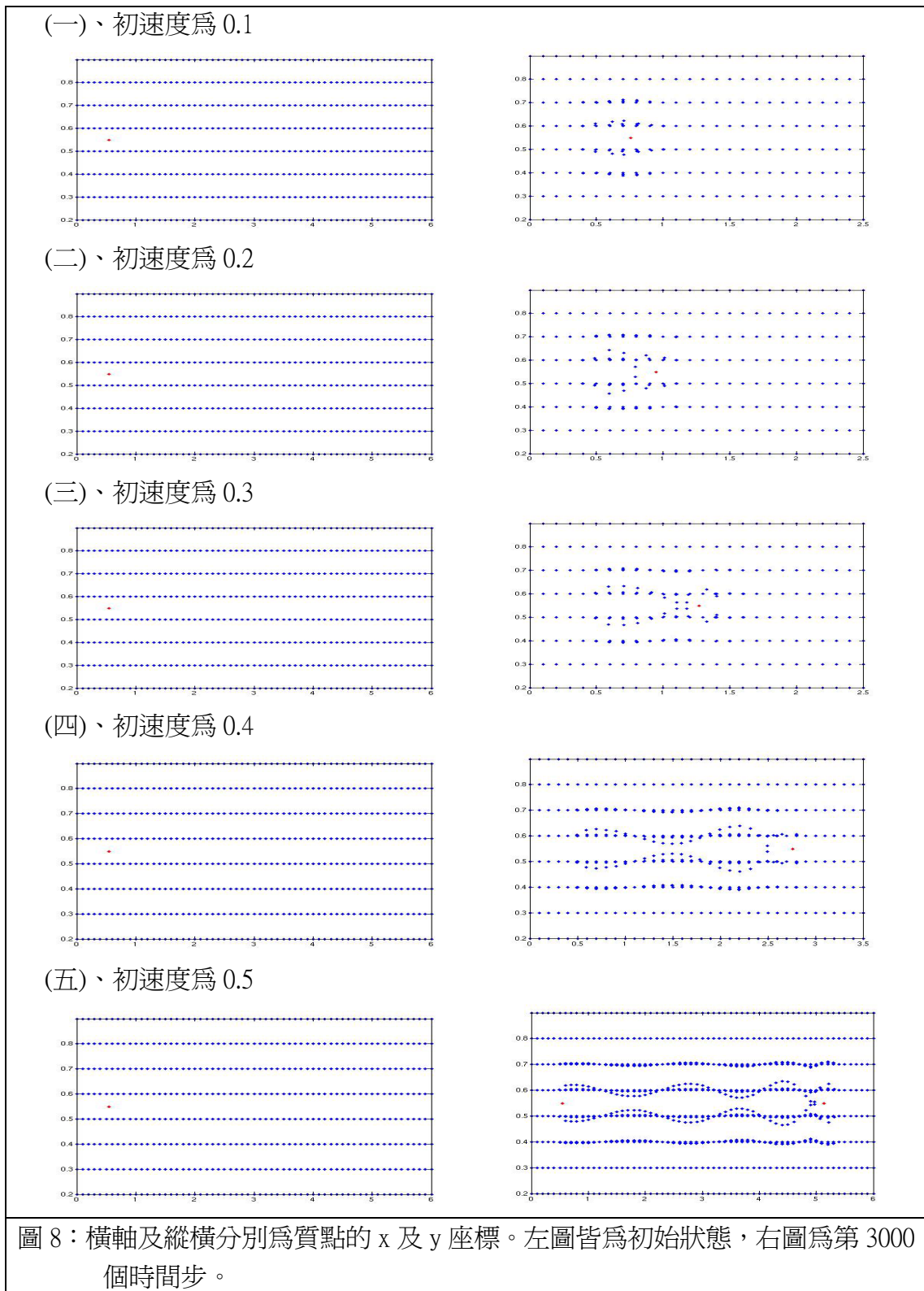


圖 7：橫軸為時間，縱軸為黑暗物質質點的動能。

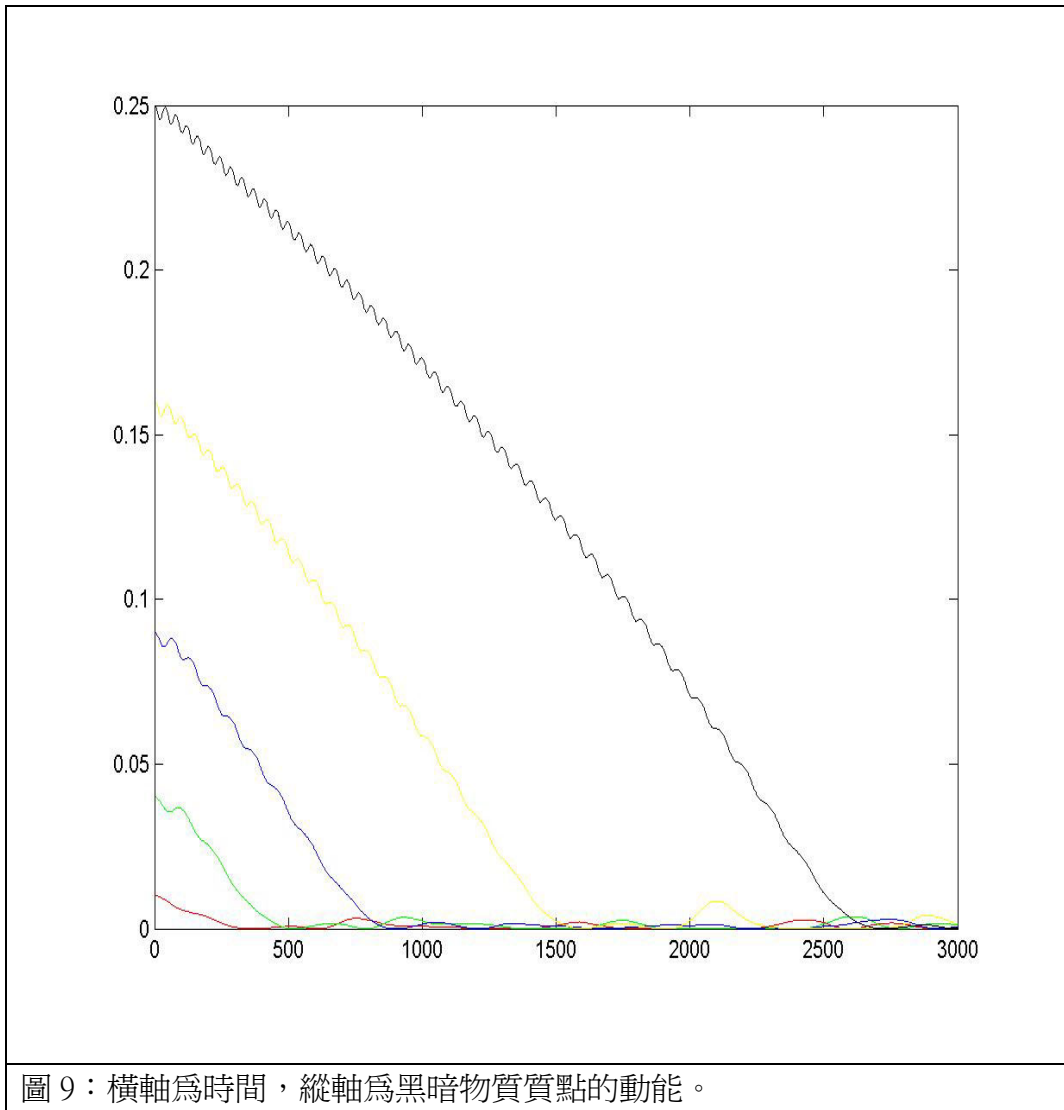
很明顯地，黑暗物質在黑暗能量中前進時，其動能  $E_k$  隨時間  $t$  作線性衰減，而且此衰減率（即圖中之線的斜率），經仔細分析為（當  $\rho_{DE} = 1$  時）

$$\frac{dE_k}{dt} = -9 \times 10^{-5}$$

四、在眾多黑暗能量中放入一個黑暗物質的質點，觀察它的運動（改變初速度）：以下每一情況中之左圖皆為初始狀態，右圖為第 3000 個時間步。



由圖 8 可以很明顯地看出，黑暗物質質點的初速越大時，於同一時間內可移動得越遠，同時可造成黑暗能量質點的上下振盪。我們可以進一步分析黑暗物質質點動能隨時間的變化，如圖 9 所示，橫軸為時間，縱軸為黑暗物質質點的動能。

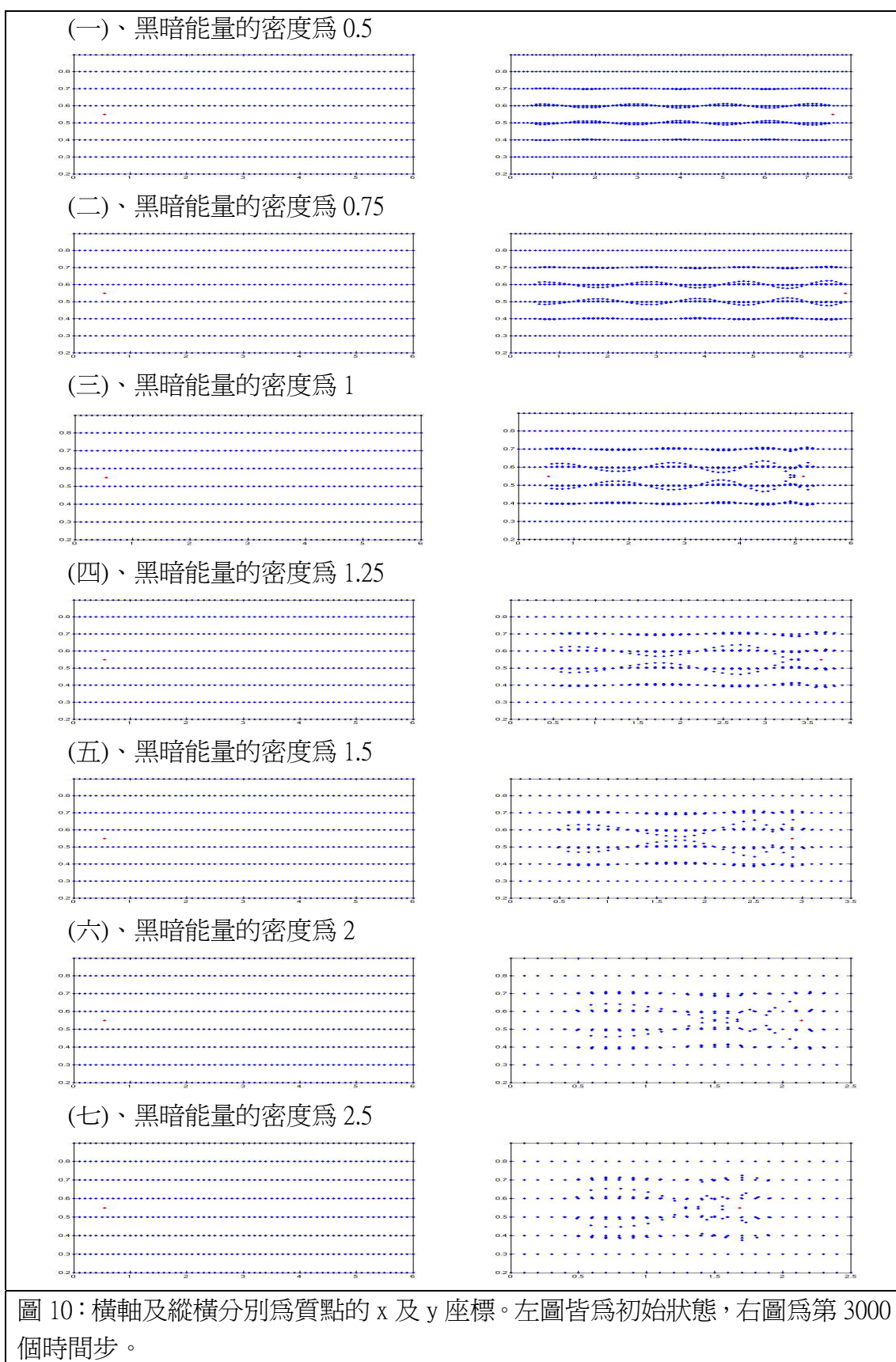


很明顯地，黑暗物質在黑暗能量中前進時，其動能  $E_k$  隨時間  $t$  作線性衰減，而且此衰減率（即圖中之線的斜率），並不會隨初速的改變而改變，經仔細分析為（當  $\rho_{DE} = 1$  時）

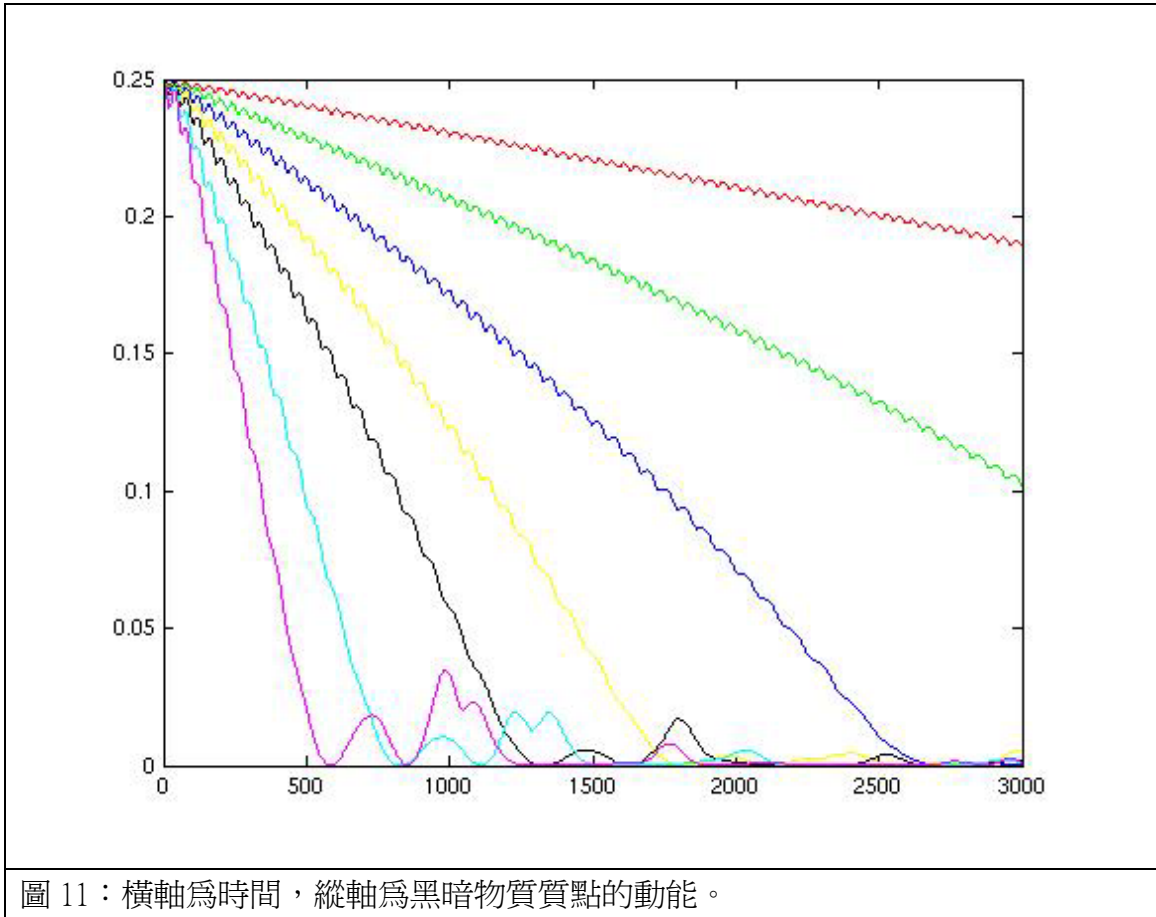
$$\frac{dE_k}{dt} = -9 \times 10^{-5} \quad (\text{公式 8})$$

圖中左下方的線之斜率和其它線不大一樣，而且圖中的每一條線上都有小幅的上下振盪，這都是因為在電腦模擬中，黑暗能量的分布是離散的質點，而不是連續的。其細節請詳見之後的討論。

五、在眾多黑暗能量中放入一個黑暗物質的質點，觀察它的運動（改變黑暗能量的密度  $\rho_{DE}$ ，但固定初速）：



由圖 10 可以很明顯地看出，黑暗能量的密度  $\rho_{DE}$  越大時，黑暗物質質點於同一時間內可移動的距離就越短，且黑暗能量質點上下振盪的振幅就越大。我們可以進一步分析黑暗物質質點動能隨時間的變化，如圖 11 所示，橫軸為時間，縱軸為黑暗物質質點的動能。



很明顯地，黑暗物質在黑暗能量中前進時，其動能  $E_k$  隨時間  $t$  作線性衰減，而且此衰減率（即圖中之線的斜率之絕對值），會隨黑暗能量密度  $\rho_{DE}$  的增加而增加，經仔細分析為

$$\frac{dE_k}{dt} \propto \rho_{DE}^{1.92} \quad (\text{公式 9})$$

圖中的線所出現的振盪，是因為在電腦模擬中，黑暗能量的分布是離散的質點，而不是連續的。其細節請詳見之後的討論。

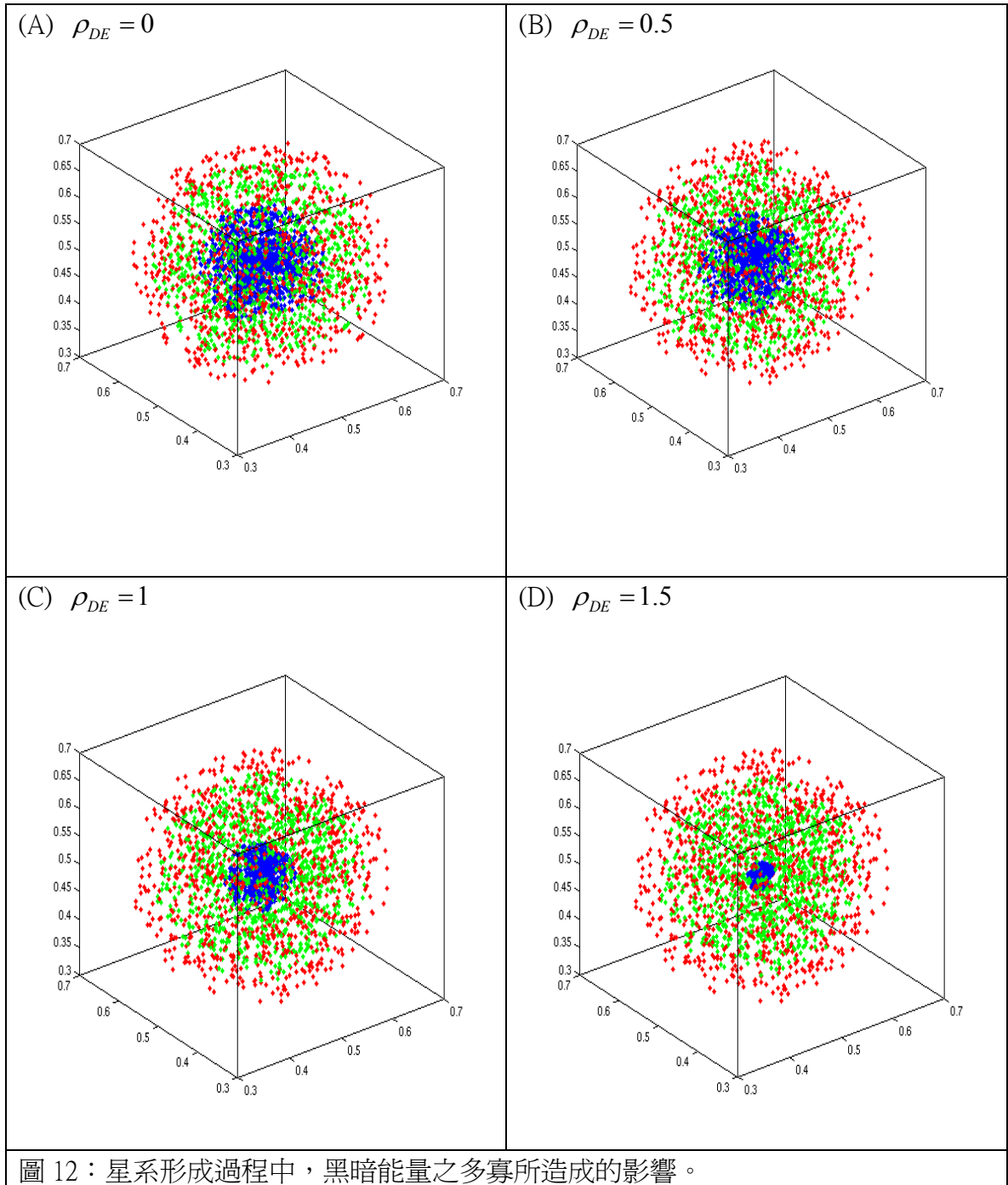
最後我們可結合（公式 8）及（公式 9）的結果，推得公式：

$$E_k(t) = E_{k0} - 9 \times 10^{-5} \rho_{DE}^{1.92} t \quad (\text{公式 10})$$

其中  $E_{k0}$  黑暗物質質點的初始動能。因此，只要給定黑暗能量的密度  $\rho_{DE}$ ，及黑暗物質質點的初始動能  $E_{k0}$ ，我們便可預測此質點的運動方式。

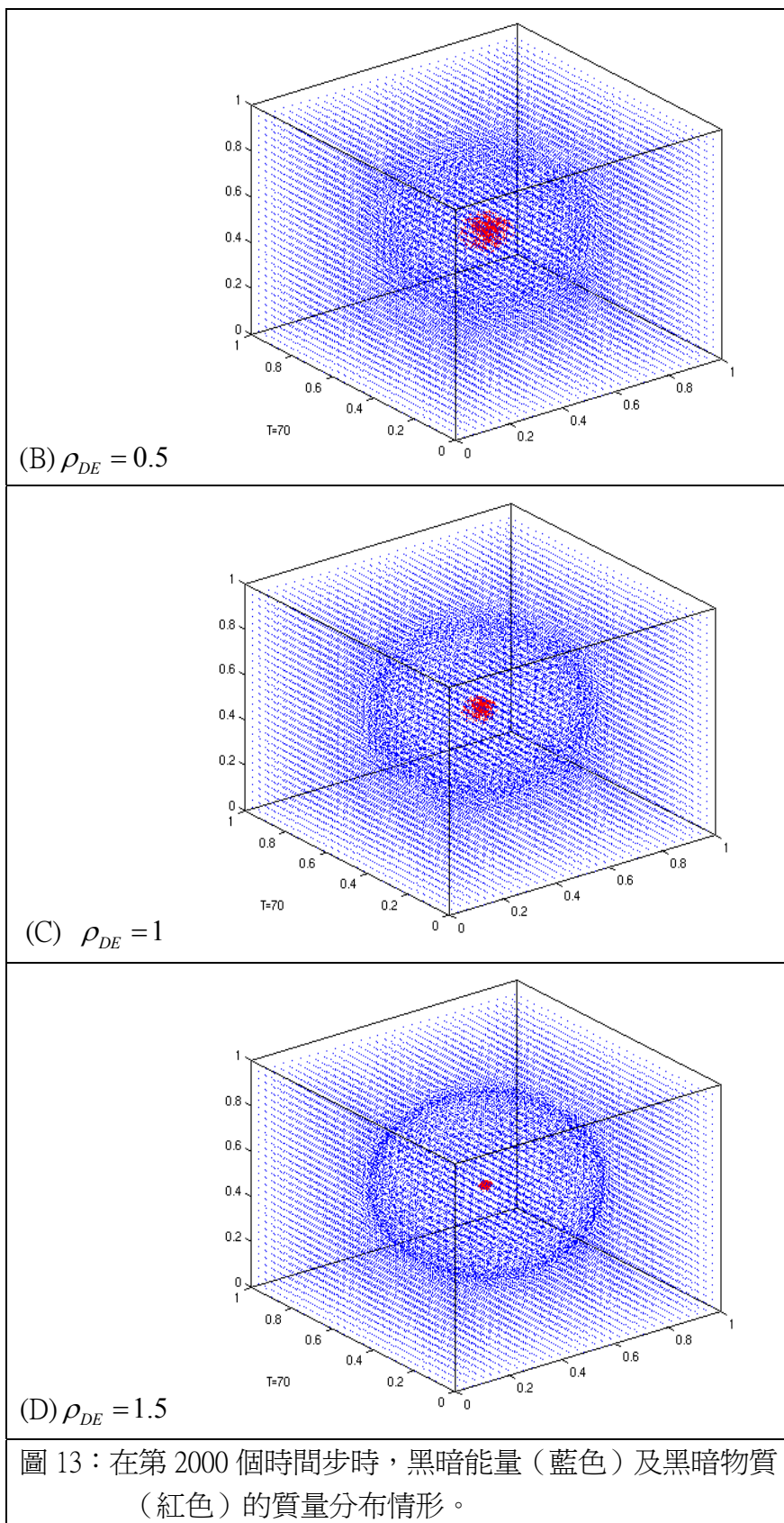
六、放置大量的黑暗物質，模擬宇宙中星系的形成：

此部分分爲兩種情形，一個爲具有黑暗能量，另一個爲沒有黑暗能量。圖 1 2 爲此兩種情形的比較結果。圖中的紅色爲起始時黑暗物質的位置，綠色爲經過 1000 個時間步時黑暗物質的位置，藍色爲經過 2000 個時間步時黑暗物質的位置。很顯然地，當黑暗能量密度  $\rho_{DE}$  越大時，會加快星系的質量往中心集中。

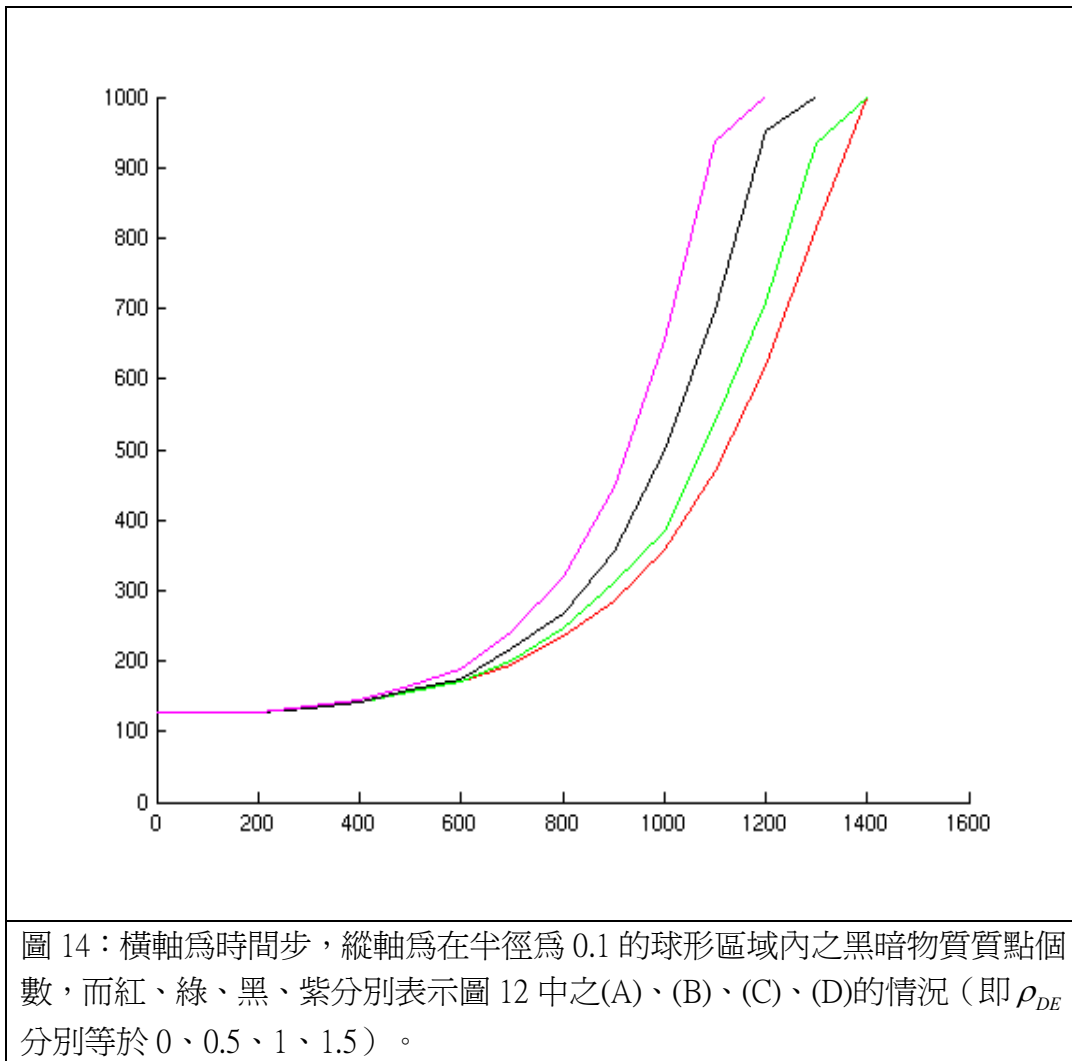


在具有黑暗能量的情況下（圖 12 中的 B、C、D），我們也將第 2000 個時間步時，黑暗能量及黑暗物質的質量分布，同時畫在一個圖中，如圖 13 所示。非常有趣

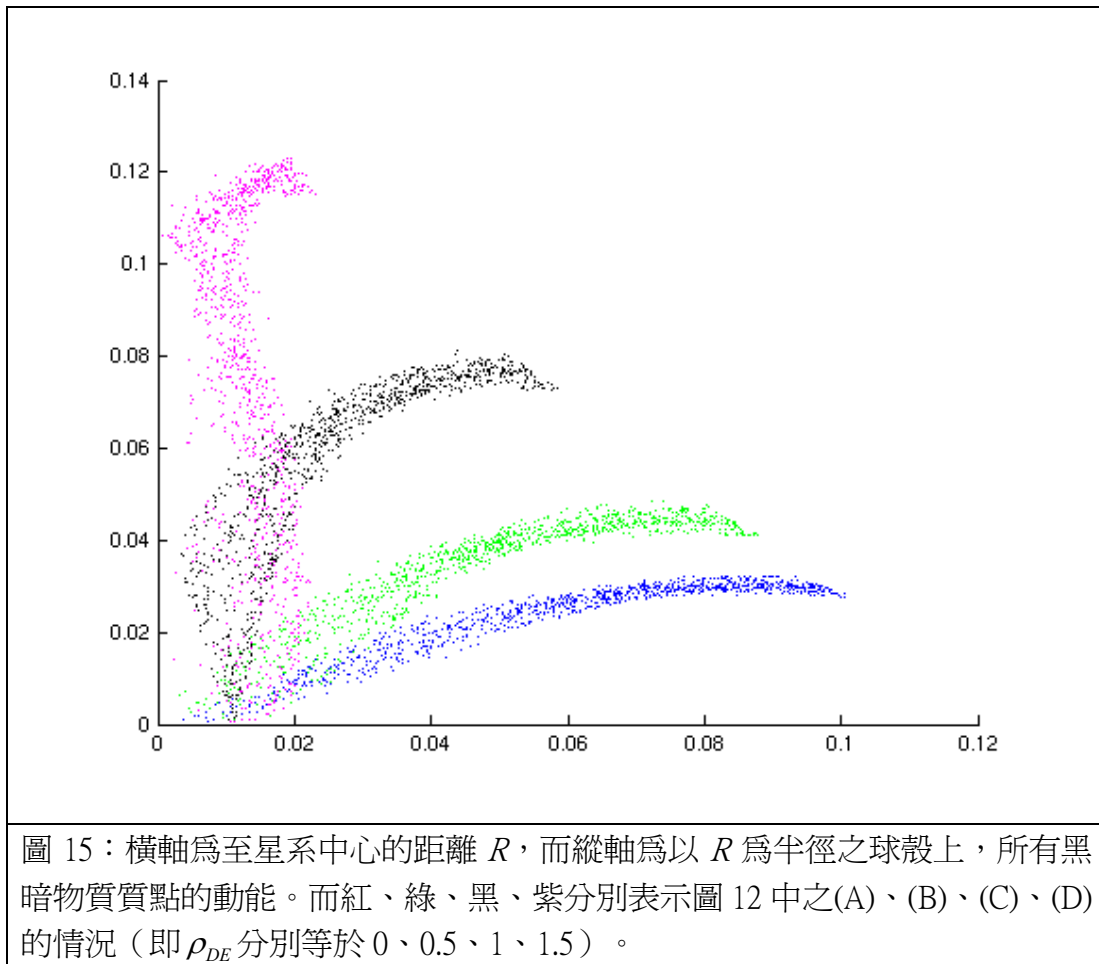
地，黑暗能量由於和黑暗物質推擠，而當後者往中央集中時，前者卻往四周散開。當黑暗能量越多時，此現象越嚴重。



爲了進一步定量地探討黑暗能量之多寡，對加速黑暗物質集中的效應，我們進行了以下的分析。在初始時間時，共有 1000 個黑暗物質質點均勻地分布在半徑爲 0.2 的球形區域內，故我們可考慮，在半徑爲 0.1 的球形區域內之黑暗物質質點個數，在不同的  $\rho_{DE}$  下，會各自如何隨時間增加。結果如圖 14 所示。橫軸爲時間步，縱軸爲在半徑爲 0.1 的球形區域內之黑暗物質質點個數，而紅、綠、黑、紫分別表示圖 10 中之(A)、(B)、(C)、(D)的情況（即  $\rho_{DE}$  分別等於 0、0.5、1、1.5）。我們可以很明顯地看出，在黑暗能量越多時，星系的核心就會越早形成。



接著，我們要探討黑暗能量之多寡，對星系結構的影響。我們可以考慮於同一時間上，在圖 12 中之(A)、(B)、(C)、(D)的情況下（即  $\rho_{DE}$  分別等於 0、0.5、1、1.5），星系中距離中心  $R$  的球殼上之平均動能隨  $R$  的變化，如圖 15 所示。其中橫軸爲至星系中心的距離  $R$ ，而縱軸爲以  $R$  爲半徑之球殼上，所有黑暗物質質點的動能  $E_k$ 。非常令人驚訝地，在黑暗能量密度  $\rho_{DE}$  不同時，此  $E_k(R)$  函數具有完全不同的結構，因此在觀測上，我們可以試圖去量測此  $E_k(R)$  函數，然後和這裡的電腦模擬結果比對，我們就可以推出宇宙中黑暗能量的密度  $\rho_{DE}$ 。



## 伍、討論

- 一、在圖 9 及圖 11 中的線，皆呈小幅度的振盪，這些上下振盪是由於電腦模擬中，黑暗能量的分佈是離散的，而不是呈現連續的分佈，因此，當一顆黑暗物質的粒子在接近一顆黑暗能量粒子時，會因推力而減速，反之，當它通過此黑暗能量粒子後而遠離時，它會因推力而加速。所以，黑暗物質的粒子在行進的過程中，會不停地遭遇到這樣的過程，以致不停地小幅減及加速，而造成圖 9 及圖 11 中的曲線上下振盪。因此，只要不斷的將電腦模擬的解析度提高（也就是將黑暗能量粒子的單位質量及彼此距離都不斷縮小），這些振盪的現象就會發生在越高頻且振幅減小，但這將會需要更多的時間作電腦計算。此處我們已經過不斷的嘗試，找出不使結果太失真，又可在合理的時間內計算出結果的解析度，約為箱子每邊長各有 31 顆黑暗能量粒子，故共有  $31^3 = 29791$  顆黑暗能量粒子。黑暗物質的粒子數則為 1000 顆。
- 二、在圖 9、圖 11 及（公式 10）中，我們觀察到，黑暗物質粒子的動能，會隨時間的增加而減少，這個現象可用**能量守恒定理**來解釋。在圖 8 及圖 10 中，我們看到黑暗物質粒子在向右方前進時，會造成黑暗能量粒子沿垂直於行進的方向上振盪，

由原本的靜止而變成具有振盪的力學能，故依力學能守恆定律，由於黑暗能量獲得能量，黑暗物必然會損失能量，而致使進行速率減緩。

- 三、在圖 13 中，當黑暗物質往中央聚集形成星系時，黑暗能量會被往四週推開。這是很自然的**作用力與反作用力定律**（牛頓第三運動定律）之結果，也就是說，當黑暗能量被黑暗物質往四週推開時，黑暗物質所受到的反作用力，會致使黑暗物質更往中心集中，因此造成圖 12 及 13 中，黑暗能量越多、黑暗物質就越集中的現象。
- 四、在計算圖 12 及 13 的過程中，需要計算兩粒子間的交互作用力。我們只考慮距離為 0.25 以內的粒子間之交互作用，而捨棄更長距離之粒子間的交互作用力，這是為了模擬宇宙是無窮大的事實。因為如果不這麼做，所有的粒子都會受到一個來自它們總質心（位於箱子中心）的淨力，一旦黑暗能量較大時，所有的黑暗物質都會被向外推擠出箱子外，就如圖 3 及圖 5 所觀察到的現象一般。在真實的宇宙中，較長距離的粒子間之作用力，通常一方面由於宇宙大結構上的均勻性，另一方面由於重力的反平方定理，會變得合力幾乎為零，故可忽略不計。
- 五、在計算反平方力的過程中，我們加入了所謂的 Softening length，也就是防止在兩顆粒子距離為零時，發生力為無窮大的不合理現象。這是一般多體運動模擬中所廣泛使用的技巧。

## 陸、結論

我們藉由理論的推導，配合電腦模擬的手段，來探討宇宙中黑暗物質和黑暗能量的神祕力量。我們發現：

- 一、黑暗能量的存在，會透過交互作用而從運動中的黑暗物質獲得力學能，而且因力學能守恆致使黑暗物質減慢速率，損失動能，滿足

$$E_k = E_{k0} - 9 \times 10^{-5} \rho_{DE}^{1.92} t$$

其中  $E_k$  為黑暗物質的動能， $E_{k0}$  為其初始動能， $\rho_{DE}$  為黑暗能量的密度， $t$  為時間。

- 二、星系的形成過程及結構，會因黑暗能量的存在而改變。
- 三、在黑暗能量越多時，星系的核心就會越早形成。
- 四、在距離星系中心  $R$  上的動能  $E_k(R)$ ，會因黑暗能量密度  $\rho_{DE}$  的不同，而呈現不同的線形結構，因此在觀測上，我們可以試圖去量測此  $E_k(R)$  函數，然後和這裡的電腦模擬結果比對，就可以推得出宇宙中黑暗能量的密度  $\rho_{DE}$ 。

這些研究成果，將可以在短期的未來，直接應用在觀測結果上。

## 七、參考資料及其他

- (一) 高中課本《物質科學物理篇（上）》，龍騰文化事業公司。
- (二) Scientific American, NO.25（科學人，2004 年 3 月號；中文版），遠流出版公司。
- (三) “A first course in general relativity”，B. F. Schutz, Cambridge University Press.

# 附件

## 黑暗物質和黑暗能量密度相同

```
#include<stdio.h>
#include<stdlib.h>
#include<math.h>

#define mn 10
#define r0 1.0e-10
#define mas 2.0e-4
#define turn 200

extern double sqrt();
double tempd1, tempd2, tempd3, tempd4, tempd5, tempd6;

int main(void)
{
    int N,M,R;
    double x,y,z,deltaT,m1[mn][3],m2[mn][3],F1[mn][3],F2[mn][3],a1[mn][3],a2[mn][3],v1[mn][3],v2[mn][3];
    char buf[32],filename[64];

    deltaT=1.0e-3;
    N=0;
    while(N<mn)
    {
        x=((double)rand())/32768.0;
        y=((double)rand())/32768.0;
        z=((double)rand())/32768.0;

        if(sqrt(((x-0.5)*(x-0.5)+(y-0.5)*(y-0.5)+(z-0.5)*(z-0.5)))<=0.4)
        {
            m1[N][0]=x;
            m1[N][1]=y;
            m1[N][2]=z;
            F1[N][0]=0.0;
            F1[N][1]=0.0;
            F1[N][2]=0.0;
            a1[N][0]=0.0;
            a1[N][1]=0.0;
            a1[N][2]=0.0;
            v1[N][0]=0.0;
            v1[N][1]=0.0;
            v1[N][2]=0.0;
            N++;
        }
    }

    N=0;
    while(N<mn)
    {
        x=((double)rand())/32768.0;
        y=((double)rand())/32768.0;
        z=((double)rand())/32768.0;

        if(sqrt(((x-0.5)*(x-0.5)+(y-0.5)*(y-0.5)+(z-0.5)*(z-0.5)))<=0.4)
        {
            m2[N][0]=x;
```

```

        m2[N][1]=y;
        m2[N][2]=z;
        F2[N][0]=0.0;
        F2[N][1]=0.0;
        F2[N][2]=0.0;
        a2[N][0]=0.0;
        a2[N][1]=0.0;
        a2[N][2]=0.0;
        v2[N][0]=0.0;
        v2[N][1]=0.0;
        v2[N][2]=0.0;
        N++;
    }
}

FILE *output;
output=fopen("output0.txt","w");
for(N=0;N<mn;N++)
    fprintf(output, "%e %e %e %e %e %e\n", m1[N][0],m1[N][1],m1[N][2], v1[N][0],v1[N][1],v1[N][2]);

fprintf(output,"\n");

for(N=0;N<mn;N++)
    fprintf(output, "%e %e %e %e %e %e\n", m2[N][0],m2[N][1],m2[N][2], v2[N][0],v2[N][1],v2[N][2]);

fclose(output);

for(R=1;R<turn+1;R++)
{
    for(N=0;N<mn;N++)
    {
        for(M=N+1;M<mn;M++)
        {

            tempd1=m1[M][0]-m1[N][0];
            tempd2=m1[M][1]-m1[N][1];
            tempd3=m1[M][2]-m1[N][2];
            tempd4=tempd1*tempd1+tempd2*tempd2+tempd3*tempd3;
            tempd5=sqrt(tempd4);
            tempd6=1.0*mas / ((tempd5+r0)*(tempd5+r0));
            tempd1 *= tempd6/(tempd5+r0);
            tempd2 *= tempd6/(tempd5+r0);
            tempd3 *= tempd6/(tempd5+r0);

            F1[N][0]+= tempd1;
            F1[N][1]+= tempd2;
            F1[N][2]+= tempd3;

            F1[M][0]-= tempd1;
            F1[M][1]-= tempd2;
            F1[M][2]-= tempd3;

        }
    }

    for(N=0;N<mn;N++)
    {
        for(M=0;M<mn;M++)

```

```

{
tempd1=m2[M][0]-m1[N][0];
tempd2=m2[M][1]-m1[N][1];
tempd3=m2[M][2]-m1[N][2];
tempd4=tempd1*tempd1+tempd2*tempd2+tempd3*tempd3;
tempd5=sqrt(tempd4);
tempd6=-2.0*mas / ((tempd5+r0)*(tempd5+r0));
tempd1 *= tempd6/(tempd5+r0);
tempd2 *= tempd6/(tempd5+r0);
tempd3 *= tempd6/(tempd5+r0);

F1[N][0]+= tempd1;
F1[N][1]+= tempd2;
F1[N][2]+= tempd3;

F2[N][0]-= tempd1;
F2[N][1]-= tempd2;
F2[N][2]-= tempd3;
}

a1[N][0]=F1[N][0]/mas;
a1[N][1]=F1[N][1]/mas;
a1[N][2]=F1[N][2]/mas;

m1[N][0]+=v1[N][0]*deltaT+0.5*a1[N][0]*deltaT*deltaT;
m1[N][1]+=v1[N][1]*deltaT+0.5*a1[N][1]*deltaT*deltaT;
m1[N][2]+=v1[N][2]*deltaT+0.5*a1[N][2]*deltaT*deltaT;

v1[N][0]+=a1[N][0]*deltaT;
v1[N][1]+=a1[N][1]*deltaT;
v1[N][2]+=a1[N][2]*deltaT;
}

for(N=0;N<mn;N++)
{
for(M=N+1;M<mn;M++)
{

tempd1=m2[M][0]-m2[N][0];
tempd2=m2[M][1]-m2[N][1];
tempd3=m2[M][2]-m2[N][2];
tempd4=tempd1*tempd1+tempd2*tempd2+tempd3*tempd3;
tempd5=sqrt(tempd4);
tempd6=-2.0*mas / ((tempd5+r0)*(tempd5+r0));
tempd1 *= tempd6/(tempd5+r0);
tempd2 *= tempd6/(tempd5+r0);
tempd3 *= tempd6/(tempd5+r0);

F2[N][0]+= tempd1;
F2[N][1]+= tempd2;
F2[N][2]+= tempd3;

F2[M][0]-= tempd1;
F2[M][1]-= tempd2;
F2[M][2]-= tempd3;

}

a2[N][0]=F2[N][0]/mas;

```

```

a2[N][1]=F2[N][1]/mas;
a2[N][2]=F2[N][2]/mas;

m2[N][0]+=v2[N][0]*deltaT+0.5*a2[N][0]*deltaT*deltaT;
m2[N][1]+=v2[N][1]*deltaT+0.5*a2[N][1]*deltaT*deltaT;
m2[N][2]+=v2[N][2]*deltaT+0.5*a2[N][2]*deltaT*deltaT;

v2[N][0]+=a2[N][0]*deltaT;
v2[N][1]+=a2[N][1]*deltaT;
v2[N][2]+=a2[N][2]*deltaT;
}

strcpy(filename,"output");
strcat(filename,gcvt((float)R,4,buf));
strcat(filename,".txt");
FILE *output1;
output1=fopen(filename,"w");
for(N=0;N<mn;N++)
    fprintf(output1, "%e %e %e %e %e %e\n", m1[N][0],m1[N][1],m1[N][2],
        v1[N][0],v1[N][1],v1[N][2]);

fprintf(output1,"n");

for(N=0;N<mn;N++)
    fprintf(output1, "%e %e %e %e %e %e\n", m2[N][0],m2[N][1],m2[N][2],
        v2[N][0],v2[N][1],v2[N][2]);

fclose(output1);

for(N=0;N<mn;N++)
{
    F1[N][0]=0.0;
    F1[N][1]=0.0;
    F1[N][2]=0.0;
    a1[N][0]=0.0;
    a1[N][1]=0.0;
    a1[N][2]=0.0;
    F2[N][0]=0.0;
    F2[N][1]=0.0;
    F2[N][2]=0.0;
    a2[N][0]=0.0;
    a2[N][1]=0.0;
    a2[N][2]=0.0;
}
}
}

```

## 黑暗物質和黑暗能量密度爲 3:7

```
#include<stdio.h>
#include<stdlib.h>
#include<math.h>

#define mn 50
#define r0 1.0e-10
#define mas 1.0e-5
#define turn 200

extern double sqrt();
double tempd1, tempd2, tempd3, tempd4, tempd5, tempd6;

int main(void)
{
    int N,M,R;
    double x,y,z,deltaT,m1[mn][3],m2[mn][3],F1[mn][3],F2[mn][3],a1[mn][3],a2[mn][3],v1[mn][3],v2[mn][3];
    char buf[32],filename[64];

    deltaT=1.0e-3;
    N=0;
    while(N<mn)
    {
        x=((double)rand())/32768.0;
        y=((double)rand())/32768.0;
        z=((double)rand())/32768.0;

        if(sqrt(((x-0.5)*(x-0.5)+(y-0.5)*(y-0.5)+(z-0.5)*(z-0.5))<=0.4))
        {
            m1[N][0]=x;
            m1[N][1]=y;
            m1[N][2]=z;
            F1[N][0]=0.0;
            F1[N][1]=0.0;
            F1[N][2]=0.0;
            a1[N][0]=0.0;
            a1[N][1]=0.0;
            a1[N][2]=0.0;
            v1[N][0]=0.0;
            v1[N][1]=0.0;
            v1[N][2]=0.0;
            N++;
        }
    }

    N=0;
    while(N<mn)
    {
        x=((double)rand())/32768.0;
        y=((double)rand())/32768.0;
        z=((double)rand())/32768.0;

        if(sqrt(((x-0.5)*(x-0.5)+(y-0.5)*(y-0.5)+(z-0.5)*(z-0.5))<=0.4))
        {
            m2[N][0]=x;
            m2[N][1]=y;
```

```

        m2[N][2]=z;
        F2[N][0]=0.0;
        F2[N][1]=0.0;
        F2[N][2]=0.0;
        a2[N][0]=0.0;
        a2[N][1]=0.0;
        a2[N][2]=0.0;
        v2[N][0]=0.0;
        v2[N][1]=0.0;
        v2[N][2]=0.0;
        N++;
    }
}

FILE *output;
output=fopen("output0.txt","w");
for(N=0;N<mn;N++)
    fprintf(output, "%e %e %e %e %e %e\n", m1[N][0],m1[N][1],m1[N][2], v1[N][0],v1[N][1],v1[N][2]);

fprintf(output, "\n");

for(N=0;N<mn;N++)
    fprintf(output, "%e %e %e %e %e %e\n", m2[N][0],m2[N][1],m2[N][2], v2[N][0],v2[N][1],v2[N][2]);

fclose(output);

for(R=1;R<turn+1;R++)
{
    for(N=0;N<mn;N++)
    {
        for(M=N+1;M<mn;M++)
        {

            tempd1=m1[M][0]-m1[N][0];
            tempd2=m1[M][1]-m1[N][1];
            tempd3=m1[M][2]-m1[N][2];
            tempd4=tempd1*tempd1+tempd2*tempd2+tempd3*tempd3;
            tempd5=sqrt(tempd4);
            tempd6=1.0*mas*3 / ((tempd5+r0)*(tempd5+r0));
            tempd1 *= tempd6/(tempd5+r0);
            tempd2 *= tempd6/(tempd5+r0);
            tempd3 *= tempd6/(tempd5+r0);

            F1[N][0]+= tempd1;
            F1[N][1]+= tempd2;
            F1[N][2]+= tempd3;

            F1[M][0]-= tempd1;
            F1[M][1]-= tempd2;
            F1[M][2]-= tempd3;

        }
    }

    for(N=0;N<mn;N++)
    {
        for(M=0;M<mn;M++)
        {

```

```

tempd1=m2[M][0]-m1[N][0];
tempd2=m2[M][1]-m1[N][1];
tempd3=m2[M][2]-m1[N][2];
tempd4=tempd1*tempd1+tempd2*tempd2+tempd3*tempd3;
tempd5=sqrt(tempd4);
tempd6=-2.0*mas*7 / ((tempd5+r0)*(tempd5+r0));
tempd1 *= tempd6/(tempd5+r0);
tempd2 *= tempd6/(tempd5+r0);
tempd3 *= tempd6/(tempd5+r0);

F1[N][0]+= tempd1;
F1[N][1]+= tempd2;
F1[N][2]+= tempd3;

F2[N][0]-= tempd1;
F2[N][1]-= tempd2;
F2[N][2]-= tempd3;
}

a1[N][0]=F1[N][0]/mas;
a1[N][1]=F1[N][1]/mas;
a1[N][2]=F1[N][2]/mas;

m1[N][0]+=v1[N][0]*deltaT+0.5*a1[N][0]*deltaT*deltaT;
m1[N][1]+=v1[N][1]*deltaT+0.5*a1[N][1]*deltaT*deltaT;
m1[N][2]+=v1[N][2]*deltaT+0.5*a1[N][2]*deltaT*deltaT;

v1[N][0]+=a1[N][0]*deltaT;
v1[N][1]+=a1[N][1]*deltaT;
v1[N][2]+=a1[N][2]*deltaT;
}

for(N=0;N<mn;N++)
{
    for(M=N+1;M<mn;M++)
    {

tempd1=m2[M][0]-m2[N][0];
tempd2=m2[M][1]-m2[N][1];
tempd3=m2[M][2]-m2[N][2];
tempd4=tempd1*tempd1+tempd2*tempd2+tempd3*tempd3;
tempd5=sqrt(tempd4);
tempd6=-2.0*mas*7 / ((tempd5+r0)*(tempd5+r0));
tempd1 *= tempd6/(tempd5+r0);
tempd2 *= tempd6/(tempd5+r0);
tempd3 *= tempd6/(tempd5+r0);

F2[N][0]+= tempd1;
F2[N][1]+= tempd2;
F2[N][2]+= tempd3;

F2[M][0]-= tempd1;
F2[M][1]-= tempd2;
F2[M][2]-= tempd3;

}
}

```

```

a2[N][0]=F2[N][0]/mas;
a2[N][1]=F2[N][1]/mas;
a2[N][2]=F2[N][2]/mas;

m2[N][0]+=v2[N][0]*deltaT+0.5*a2[N][0]*deltaT*deltaT;
m2[N][1]+=v2[N][1]*deltaT+0.5*a2[N][1]*deltaT*deltaT;
m2[N][2]+=v2[N][2]*deltaT+0.5*a2[N][2]*deltaT*deltaT;

v2[N][0 ]+=a2[N][0]*deltaT;
v2[N][1]+=a2[N][1]*deltaT;
v2[N][2]+=a2[N][2]*deltaT;
}

strcpy(filename,"output");
strcat(filename,gcvt((float)R,4,buf));
strcat(filename,".txt");
FILE *output1;
output1=fopen(filename,"w");
for(N=0;N<mn;N++)
    fprintf(output1, "%e %e %e %e %e %e\n", m1[N][0],m1[N][1],m1[N][2],
        v1[N][0],v1[N][1],v1[N][2]);

fprintf(output1,"\n");

for(N=0;N<mn;N++)
    fprintf(output1, "%e %e %e %e %e %e\n", m2[N][0],m2[N][1],m2[N][2],
        v2[N][0],v2[N][1],v2[N][2]);

fclose(output1);

for(N=0;N<mn;N++)
{
    F1[N][0]=0.0;
    F1[N][1]=0.0;
    F1[N][2]=0.0;
    a1[N][0]=0.0;
    a1[N][1]=0.0;
    a1[N][2]=0.0;
    F2[N][0]=0.0;
    F2[N][1]=0.0;
    F2[N][2]=0.0;
    a2[N][0]=0.0;
    a2[N][1]=0.0;
    a2[N][2]=0.0;
}
}
}

```

## 在眾多黑暗能量中放入一個黑暗物質的質點

```
#include<stdio.h>
#include<stdlib.h>
#include<math.h>

#define mn 10
#define r0 1.0e-2
#define mas 5.0e-4
#define turn 100

extern double sqrt();
double tempd1, tempd2, tempd3, tempd4, tempd5, tempd6;

int main(void)
{
int N,M,R,a;
double
x,y,z,deltaT,m1[1][3],m2[mn*mn*mn*5][3],a1[mn*mn*mn][3],a2[mn*mn*mn*5][3],v1[mn*mn*mn][3],v2[mn*mn*
mn*5][3],r[mn*mn*mn*5][3];
char buf[32],filename[64];

deltaT=1.0e-1;

a=0;
for(N=0;N<mn*5;N++)
{
for(M=0;M<mn;M++)
{
for(R=0;R<mn;R++)
{
x=((double)N)/10.0;
y=((double)M)/10.0;
z=((double)R)/10.0;

m2[a][0]=x;
m2[a][1]=y;
m2[a][2]=z;
a2[a][0]=0.0;
a2[a][1]=0.0;
a2[a][2]=0.0;
v2[a][0]=0.0;
v2[a][1]=0.0;
v2[a][2]=0.0;
r[a][0]=0.0;
r[a][1]=0.0;
r[a][2]=0.0;

a=a+1;

}
}
}
printf("a=%d\n", a);
fflush(stdout);
N=0;
```

```

m1[0][0]=0.5;
m1[0][1]=0.55;
m1[0][2]=0.55;
v1[0][0]=0.5;
v1[0][1]=0.0;
v1[0][2]=0.0;
a1[0][0]=0.0;
a1[0][1]=0.0;
a1[0][2]=0.0;
printf("1\n");
fflush(stdout);

FILE *output, *fid;

fid=fopen("vt0.txt","w");
fprintf(fid, "%e %e %e %e %e %e\n",v1[0][0],v1[0][1],v1[0][2],m1[0][0],m1[0][1],m1[0][2]);

output=fopen("output0.txt","wb");
fwrite(&m1[0][0],sizeof(double), 3, output);
fwrite(&v1[0][0],sizeof(double), 3, output);
for(N=0;N<a;N++){
    fwrite(&m2[N][0],sizeof(double), 3, output);
    fwrite(&v2[N][0],sizeof(double), 3, output);
}
fclose(output);
printf("2\n");
fflush(stdout);

M=0;
N=0;
R=0;
for(R=1;R<turn+1;R++)
{
    for(M=0;M<a;M++)
    {
        tempd1=m2[M][0]-m1[0][0];
        tempd2=m2[M][1]-m1[0][1];
        tempd3=m2[M][2]-m1[0][2];
        tempd4=tempd1*tempd1+tempd2*tempd2+tempd3*tempd3;
        tempd5=sqrt(tempd4);
        if(tempd5<=0.25)
        {
            tempd6=-2.0*mas/((tempd5+r0)*(tempd5+r0));
            tempd1 *= tempd6/(tempd5+r0);
            tempd2 *= tempd6/(tempd5+r0);
            tempd3 *= tempd6/(tempd5+r0);

            a1[0][0]+= tempd1;
            a1[0][1]+= tempd2;
            a1[0][2]+= tempd3;

            a2[M][0]-= tempd1;
            a2[M][1]-= tempd2;
            a2[M][2]-= tempd3;
        }
    }
    m1[0][0]+=v1[0][0]*deltaT+0.5*a1[0][0]*deltaT*deltaT;
    m1[0][1]+=v1[0][1]*deltaT+0.5*a1[0][1]*deltaT*deltaT;
    m1[0][2]+=v1[0][2]*deltaT+0.5*a1[0][2]*deltaT*deltaT;
}

```

```

v1[0][0]+=a1[0][0]*deltaT;
v1[0][1]+=a1[0][1]*deltaT;
v1[0][2]+=a1[0][2]*deltaT;

for(N=0;N<a;N++)
{
    for(M=N+1;M<a;M++)
    {
        tempd1=m2[M][0]-m2[N][0];
        if (tempd1>4.0) tempd1 -= 5.0;
        if (tempd1<-4.0) tempd1 += 5.0;
        tempd2=m2[M][1]-m2[N][1];
        if (tempd2>0.7) tempd2 -= 1.0;
        if (tempd2<-0.7) tempd2 += 1.0;
        tempd3=m2[M][2]-m2[N][2];
        if (tempd3>0.7) tempd3 -= 1.0;
        if (tempd3<-0.7) tempd3 += 1.0;
        tempd4=tempd1*tempd1+tempd2*tempd2+tempd3*tempd3;
        tempd5=sqrt(tempd4);
        if(tempd5<=0.25)
        {
            tempd6=-2.0*mas / ((tempd5+r0)*(tempd5+r0));
            tempd1 *= tempd6/(tempd5+r0);
            tempd2 *= tempd6/(tempd5+r0);
            tempd3 *= tempd6/(tempd5+r0);

            a2[N][0]+= tempd1;
            a2[N][1]+= tempd2;
            a2[N][2]+= tempd3;

            a2[M][0]-= tempd1;
            a2[M][1]-= tempd2;
            a2[M][2]-= tempd3;
        }
    }
    m2[N][0]+=v2[N][0]*deltaT+0.5*a2[N][0]*deltaT*deltaT;
    m2[N][1]+=v2[N][1]*deltaT+0.5*a2[N][1]*deltaT*deltaT;
    m2[N][2]+=v2[N][2]*deltaT+0.5*a2[N][2]*deltaT*deltaT;

    v2[N][0]+=a2[N][0]*deltaT;
    v2[N][1]+=a2[N][1]*deltaT;
    v2[N][2]+=a2[N][2]*deltaT;
}
if(R%10==0){
    strcpy(filename,"output");
    strcat(filename,gcvt((float)R,4,buf));
    strcat(filename,".txt");
    FILE *output1;
    output1=fopen(filename,"wb");
    fwrite(&m1[0][0],sizeof(double), 3, output1);
    fwrite(&v1[0][0],sizeof(double), 3, output1);
    for(N=0;N<a;N++){
        fwrite(&m2[N][0],sizeof(double), 3, output1);
        fwrite(&v2[N][0],sizeof(double), 3, output1);
    }
    fclose(output1);
}
}

```

```
fprintf(fid, "%d %e %e %e %e %e %e\n",R,v1[0][0],v1[0][1],v1[0][2],m1[0][0],m1[0][1],m1[0][2]);  
fflush(fid);
```

```
a1[0][0]=0.0;  
a1[0][1]=0.0;  
a1[0][2]=0.0;  
for(N=0;N<a;N++)  
{  
    a2[N][0]=0.0;  
    a2[N][1]=0.0;  
    a2[N][2]=0.0;  
}
```

```
}  
printf("a1=%e\n", a1[0][0]);  
printf("a1=%e\n", a1[0][1]);  
printf("a1=%e\n", a1[0][2]);  
fflush(stdout);  
fclose(fid);  
}
```

## 放置一顆黑暗物質並改變初速度

```
#include<stdio.h>
#include<stdlib.h>
#include<math.h>

#define mn 10
#define r0 1.0e-10
#define mas 2.0e-4
#define turn 3000

extern double sqrt();
double tempd1, tempd2, tempd3, tempd4, tempd5, tempd6;

int main(void)
{
    int N,M,R,a;
    double
    x,y,z,deltaT,m1[1][3],m2[mn*mn*mn*50][3],a1[mn*mn*mn][3],a2[mn*mn*mn*50][3],v1[mn*mn*mn][3],v2[
    mn*mn*mn*50][3],r[mn*mn*mn*50][3];
    char buf[32],filename[64];

    deltaT=5.0e-3;
    a=0;
    for(N=0;N<mn*50;N++)
    {
        for(M=0;M<mn;M++)
        {
            for(R=0;R<mn;R++)
            {
                x=((double)N)/10.0;
                y=((double)M)/10.0;
                z=((double)R)/10.0;

                m2[a][0]=x;
                m2[a][1]=y;
                m2[a][2]=z;
                a2[a][0]=0.0;
                a2[a][1]=0.0;
                a2[a][2]=0.0;
                v2[a][0]=0.0;
                v2[a][1]=0.0;
                v2[a][2]=0.0;
                r[a][0]=0.0;
                r[a][1]=0.0;
                r[a][2]=0.0;

                a=a+1;
            }
        }
    }
    N=0;

    m1[0][0]=0.55;
    m1[0][1]=0.55;
    m1[0][2]=0.5;
    v1[0][0]=初速度;
    v1[0][1]=0.0;
```

```

v1[0][2]=0.0;

FILE *output, *fid;
fid=fopen("vt0.txt","w");
fprintf(fid, "0 %e %e %e %e %e %e\n",v1[0][0],v1[0][1],v1[0][2],m1[0][0],m1[0][1],m1[0][2]);

output=fopen("output0.txt","wb");
fwrite(&m1[0][0],sizeof(double), 3, output);
fwrite(&v1[0][0],sizeof(double), 3, output);
for(N=0;N<a;N++){
    fwrite(&m2[N][0],sizeof(double), 3, output);
    fwrite(&v2[N][0],sizeof(double), 3, output);
}
fclose(output);
M=0;
N=0;
R=0;
for(R=1;R<turn+1;R++)
{
    for(M=0;M<a;M++)
    {

        tempd1=m2[M][0]-m1[0][0];
        tempd2=m2[M][1]-m1[0][1];
        tempd3=m2[M][2]-m1[0][2];
        tempd4=tempd1*tempd1+tempd2*tempd2+tempd3*tempd3;
        tempd5=sqrt(tempd4);
        if(tempd5<=0.25)
        {
            tempd6=-2.0*mas/((tempd5+r0)*(tempd5+r0));
            tempd1 *= tempd6/(tempd5+r0);
            tempd2 *= tempd6/(tempd5+r0);
            tempd3 *= tempd6/(tempd5+r0);

            a1[0][0]+= tempd1;
            a1[0][1]+= tempd2;
            a1[0][2]+= tempd3;

            a2[M][0]-= tempd1;
            a2[M][1]-= tempd2;
            a2[M][2]-= tempd3;
        }
    }

    m1[0][0]+=v1[0][0]*deltaT+0.5*a1[0][0]*deltaT*deltaT;
    m1[0][1]+=v1[0][1]*deltaT+0.5*a1[0][1]*deltaT*deltaT;
    m1[0][2]+=v1[0][2]*deltaT+0.5*a1[0][2]*deltaT*deltaT;

    v1[0][0]+=a1[0][0]*deltaT;
    v1[0][1]+=a1[0][1]*deltaT;
    v1[0][2]+=a1[0][2]*deltaT;

    for(N=0;N<a;N++)
    {
        a2[N][0]-=r[N][0]*(8.0*1.20206)*mas/0.001;
        a2[N][1]-=r[N][1]*(8.0*1.20206)*mas/0.001;
        a2[N][2]-=r[N][2]*(8.0*1.20206)*mas/0.001;

        v2[N][0]+=a2[N][0]*deltaT;

```

```

v2[N][1]+=a2[N][1]*deltaT;
v2[N][2]+=a2[N][2]*deltaT;

tempd1=v2[N][0]*deltaT+0.5*a2[N][0]*deltaT*deltaT;
tempd2=v2[N][1]*deltaT+0.5*a2[N][1]*deltaT*deltaT;
tempd3=v2[N][2]*deltaT+0.5*a2[N][2]*deltaT*deltaT;

r[N][0]+=tempd1;
r[N][1]+=tempd2;
r[N][2]+=tempd3;

m2[N][0]+=tempd1;
m2[N][1]+=tempd2;
m2[N][2]+=tempd3;
}
if(R%50==0){
strcpy(filename,"output");
strcat(filename,gcvt((float)R,4,buf));
strcat(filename,".txt");
FILE *output1;
output1=fopen(filename,"wb");
fwrite(&m1[0][0],sizeof(double), 3, output1);
fwrite(&v1[0][0],sizeof(double), 3, output1);
for(N=0;N<a;N++){
    fwrite(&m2[N][0],sizeof(double), 3, output1);
    fwrite(&v2[N][0],sizeof(double), 3, output1);
}
fclose(output1);
}

fprintf(fid, "%d %e %e %e %e %e %e\n",R,v1[0][0],v1[0][1],v1[0][2],m1[0][0],m1[0][1],m1[0][2]);

a1[0][0]=0.0;
a1[0][1]=0.0;
a1[0][2]=0.0;
for(N=0;N<a;N++)
{
    a2[N][0]=0.0;
    a2[N][1]=0.0;
    a2[N][2]=0.0;
}
}
fclose(fid);
}

```

## 放置一顆黑暗物質並改變初速度

```
#include<stdio.h>
#include<stdlib.h>
#include<math.h>

#define mn 10
#define r0 1.0e-10
#define mas 2.0e-4
#define turn 3000

extern double sqrt();
double tempd1, tempd2, tempd3, tempd4, tempd5, tempd6;

int main(void)
{
    int N,M,R,a;
    double
x,y,z,deltaT,m1[1][3],m2[mn*mn*mn*50][3],a1[mn*mn*mn][3],a2[mn*mn*mn*50][3],v1[mn*mn*mn][3],v2[mn*mn
*mn*50][3],r[mn*mn*mn*50][3];
    char buf[32],filename[64];

    deltaT=5.0e-3;
    a=0;
    for(N=0;N<mn*50;N++)
    {
        for(M=0;M<mn;M++)
        {
            for(R=0;R<mn;R++)
            {
                x=((double)N)/10.0;
                y=((double)M)/10.0;
                z=((double)R)/10.0;

                m2[a][0]=x;
                m2[a][1]=y;
                m2[a][2]=z;
                a2[a][0]=0.0;
                a2[a][1]=0.0;
                a2[a][2]=0.0;
                v2[a][0]=0.0;
                v2[a][1]=0.0;
                v2[a][2]=0.0;
                r[a][0]=0.0;
                r[a][1]=0.0;
                r[a][2]=0.0;

                a=a+1;
            }
        }
    }
    printf("a=%d\n", a);
    fflush(stdout);
    N=0;
    m1[0][0]=0.55;
    m1[0][1]=0.55;
    m1[0][2]=0.5;
    v1[0][0]=0.5;
```

```

v1[0][1]=0.0;
v1[0][2]=0.0;
a1[0][0]=0.0;
a1[0][1]=0.0;
a1[0][2]=0.0;

FILE *output, *fid;
fid=fopen("vt0.txt","w");
fprintf(fid, "0 %e %e %e %e %e %e\n",v1[0][0],v1[0][1],v1[0][2],m1[0][0],m1[0][1],m1[0][2]);

output=fopen("output0.txt","wb");
fwrite(&m1[0][0],sizeof(double), 3, output);
fwrite(&v1[0][0],sizeof(double), 3, output);
for(N=0;N<a;N++){
    fwrite(&m2[N][0],sizeof(double), 3, output);
    fwrite(&v2[N][0],sizeof(double), 3, output);
}
fclose(output);

M=0;
N=0;
R=0;
for(R=1;R<turn+1;R++)
{
    for(M=0;M<a;M++)
    {
        tempd1=m2[M][0]-m1[0][0];
        tempd2=m2[M][1]-m1[0][1];
        tempd3=m2[M][2]-m1[0][2];
        tempd4=tempd1*tempd1+tempd2*tempd2+tempd3*tempd3;
        tempd5=sqrt(tempd4);
        if(tempd5<=0.25)
        {
            tempd6=黑暗能量的密度*mas/((tempd5+r0)*(tempd5+r0));
            tempd1 *= tempd6/(tempd5+r0);
            tempd2 *= tempd6/(tempd5+r0);
            tempd3 *= tempd6/(tempd5+r0);

            a1[0][0]+= tempd1;
            a1[0][1]+= tempd2;
            a1[0][2]+= tempd3;

            a2[M][0]-= tempd1;
            a2[M][1]-= tempd2;
            a2[M][2]-= tempd3;
        }
    }

    m1[0][0]+=v1[0][0]*deltaT+0.5*a1[0][0]*deltaT*deltaT;
    m1[0][1]+=v1[0][1]*deltaT+0.5*a1[0][1]*deltaT*deltaT;
    m1[0][2]+=v1[0][2]*deltaT+0.5*a1[0][2]*deltaT*deltaT;

    v1[0][0]+=a1[0][0]*deltaT;
    v1[0][1]+=a1[0][1]*deltaT;
    v1[0][2]+=a1[0][2]*deltaT;

    for(N=0;N<a;N++)
    {
        a2[N][0]-=r[N][0]*(8.0*1.20206)*mas/0.001;
    }
}

```

```

a2[N][1]=r[N][1]*(8.0*1.20206)*mas/0.001;
a2[N][2]=r[N][2]*(8.0*1.20206)*mas/0.001;

v2[N][0]=a2[N][0]*deltaT;
v2[N][1]=a2[N][1]*deltaT;
v2[N][2]=a2[N][2]*deltaT;

tempd1=v2[N][0]*deltaT+0.5*a2[N][0]*deltaT*deltaT;
tempd2=v2[N][1]*deltaT+0.5*a2[N][1]*deltaT*deltaT;
tempd3=v2[N][2]*deltaT+0.5*a2[N][2]*deltaT*deltaT;

r[N][0]=tempd1;
r[N][1]=tempd2;
r[N][2]=tempd3;

m2[N][0]=tempd1;
m2[N][1]=tempd2;
m2[N][2]=tempd3;

}
if(R%50==0){
strcpy(filename,"output");
strcat(filename,gcvt((float)R,4,buf));
strcat(filename,".txt");
FILE *output1;
output1=fopen(filename,"wb");
fwrite(&m1[0][0],sizeof(double), 3, output1);
fwrite(&v1[0][0],sizeof(double), 3, output1);
for(N=0;N<a;N++){
    fwrite(&m2[N][0],sizeof(double), 3, output1);
    fwrite(&v2[N][0],sizeof(double), 3, output1);
}
fclose(output1);
}

fprintf(fid, "%d %e %e %e %e %e %e\n",R,v1[0][0],v1[0][1],v1[0][2],m1[0][0],m1[0][1],m1[0][2]);

a1[0][0]=0.0;
a1[0][1]=0.0;
a1[0][2]=0.0;
for(N=0;N<a;N++)
{
    a2[N][0]=0.0;
    a2[N][1]=0.0;
    a2[N][2]=0.0;
}
}
fclose(fid);
}

```

# 放置大量的黑暗物質，模擬宇宙中星系的形成

## 1. 有黑暗能量

```
#include<stdio.h>
#include<stdlib.h>
#include<math.h>

#define dm 1000
#define mn 31
#define r0 5.0e-2
#define mas 1.0e-5
#define turn 100

extern double sqrt();
double tempd1, tempd2, tempd3, tempd4, tempd5, tempd6;

int main(void)
{
    int N,M,R,a;
    double
x,y,z,deltaT,m1[dm][3],m2[mn*mn*mn][3],a1[dm][3],a2[mn*mn*mn][3],v1[dm][3],v2[mn*mn*mn][3],r[mn*mn*mn]
[3];
    char buf[32],filename[64];

    deltaT=2.0e-2;
    a=0;
    for(N=0;N<mn;N++)
    {
        for(M=0;M<mn;M++)
        {
            for(R=0;R<mn;R++)
            {
                x=((double)N)/31.0;
                y=((double)M)/31.0;
                z=((double)R)/31.0;

                m2[a][0]=x;
                m2[a][1]=y;
                m2[a][2]=z;
                a2[a][0]=0.0;
                a2[a][1]=0.0;
                a2[a][2]=0.0;
                v2[a][0]=0.0;
                v2[a][1]=0.0;
                v2[a][2]=0.0;
                r[a][0]=0.0;
                r[a][1]=0.0;
                r[a][2]=0.0;

                a=a+1;
            }
        }
    }
    printf("a=%d\n", a);
    fflush(stdout);
    N=0;
```

```

while(N<dm)
{
    x=((double)rand())/32768.0;
    y=((double)rand())/32768.0;
    z=((double)rand())/32768.0;

    if(sqrt((x-0.5)*(x-0.5)+(y-0.5)*(y-0.5)+(z-0.5)*(z-0.5))<=0.2)
    {
        m1[N][0]=x;
        m1[N][1]=y;
        m1[N][2]=z;
        a1[N][0]=0.0;
        a1[N][1]=0.0;
        a1[N][2]=0.0;
        v1[N][0]=0.0;
        v1[N][1]=0.0;
        v1[N][2]=0.0;
        N++;
    }
}

FILE *output;

output=fopen("output0.txt","wb");
for(N=0;N<dm;N++){
    fwrite(&m1[N][0],sizeof(double), 3, output);
    fwrite(&v1[N][0],sizeof(double), 3, output);
}
for(N=0;N<a;N++){
    fwrite(&m2[N][0],sizeof(double), 3, output);
    fwrite(&v2[N][0],sizeof(double), 3, output);
}
fclose(output);

M=0;
N=0;
R=0;
for(R=1;R<turn+1;R++)
{
    for(N=0;N<dm;N++)
    {
        for(M=N+1;M<dm;M++)
        {
            tempd1=m1[M][0]-m1[N][0];
            tempd2=m1[M][1]-m1[N][1];
            tempd3=m1[M][2]-m1[N][2];
            tempd4=tempd1*tempd1+tempd2*tempd2+tempd3*tempd3;
            tempd5=sqrt(tempd4);
            tempd6=1.0*mas / ((tempd5+r0)*(tempd5+r0));
            tempd1 *= tempd6/(tempd5+r0);
            tempd2 *= tempd6/(tempd5+r0);
            tempd3 *= tempd6/(tempd5+r0);

            a1[N][0]+= tempd1;
            a1[N][1]+= tempd2;
            a1[N][2]+= tempd3;

            a1[M][0]-= tempd1;

```

```

        a1[M][1]-= tempd2;
        a1[M][2]-= tempd3;
    }
}

for(N=0;N<dm;N++)
{
    for(M=0;M<a;M++)
    {
        tempd1=m2[M][0]-m1[N][0];
        tempd2=m2[M][1]-m1[N][1];
        tempd3=m2[M][2]-m1[N][2];
        tempd4=tempd1*tempd1+tempd2*tempd2+tempd3*tempd3;
        tempd5=sqrt(tempd4);
        if(tempd5<=0.25)
        {
            tempd6=黑暗能量的密度*mas/((tempd5+r0)*(tempd5+r0))/10.0;
            tempd1 *= tempd6/(tempd5+r0);
            tempd2 *= tempd6/(tempd5+r0);
            tempd3 *= tempd6/(tempd5+r0);

            a1[N][0]+= tempd1;
            a1[N][1]+= tempd2;
            a1[N][2]+= tempd3;

            a2[M][0]-= tempd1;
            a2[M][1]-= tempd2;
            a2[M][2]-= tempd3;
        }
    }
    m1[N][0]+=v1[N][0]*deltaT+0.5*a1[N][0]*deltaT*deltaT;
    m1[N][1]+=v1[N][1]*deltaT+0.5*a1[N][1]*deltaT*deltaT;
    m1[N][2]+=v1[N][2]*deltaT+0.5*a1[N][2]*deltaT*deltaT;
    v1[N][0]+=a1[N][0]*deltaT;
    v1[N][1]+=a1[N][1]*deltaT;
    v1[N][2]+=a1[N][2]*deltaT;
}
for(N=0;N<a;N++){
    a2[N][0]=-r[N][0]*(8.0*1.20206)*mas/1.0;
    a2[N][1]=-r[N][1]*(8.0*1.20206)*mas/1.0;
    a2[N][2]=-r[N][2]*(8.0*1.20206)*mas/1.0;
    v2[N][0]+=a2[N][0]*deltaT;
    v2[N][1]+=a2[N][1]*deltaT;
    v2[N][2]+=a2[N][2]*deltaT;

    tempd1=v2[N][0]*deltaT+0.5*a2[N][0]*deltaT*deltaT;
    tempd2=v2[N][1]*deltaT+0.5*a2[N][1]*deltaT*deltaT;
    tempd3=v2[N][2]*deltaT+0.5*a2[N][2]*deltaT*deltaT;

    r[N][0]+=tempd1;
    r[N][1]+=tempd2;
    r[N][2]+=tempd3;

    m2[N][0]+=tempd1;
    m2[N][1]+=tempd2;
    m2[N][2]+=tempd3;
}
if(R%5==0){
    strcpy(filename,"output");
}

```

```

        strcat(filename,gcvt((float)R,4,buf));
    strcat(filename,".txt");
    FILE *output1;
    output1=fopen(filename,"wb");
    for(N=0;N<dm;N++){
        fwrite(&m1[N][0],sizeof(double), 3, output1);
        fwrite(&v1[N][0],sizeof(double), 3, output1);
    }
    for(N=0;N<a;N++){
        fwrite(&m2[N][0],sizeof(double), 3, output1);
        fwrite(&v2[N][0],sizeof(double), 3, output1);
    }
    fclose(output1);
}

for(N=0;N<dm;N++)
{
    a1[N][0]=0.0;
    a1[N][1]=0.0;
    a1[N][2]=0.0;
}
for(N=0;N<a;N++)
{
    a2[N][0]=0.0;
    a2[N][1]=0.0;
    a2[N][2]=0.0;
}
}
}

```

## 2. 沒有黑暗能量

```

#include<stdio.h>
#include<stdlib.h>
#include<math.h>

#define dm 1000
#define mn 31
#define r0 5.0e-2
#define mas 1.0e-5
#define turn 100

extern double sqrt();
double tempd1, tempd2, tempd3, tempd4, tempd5, tempd6;

int main(void)
{
    int N,M,R,a;
    double
x,y,z,deltaT,m1[dm][3],m2[mn*mn*mn][3],a1[dm][3],a2[mn*mn*mn][3],v1[dm][3],v2[mn*mn*mn][3],r[mn*mn*mn]
[3];
    char buf[32],filename[64];
    deltaT=2.0e-2;
    a=0;
    for(N=0;N<mn;N++)
    {

```

```

for(M=0;M<mn;M++)
{
    for(R=0;R<mn;R++)
    {
        x=((double)N)/31.0;
        y=((double)M)/31.0;
        z=((double)R)/31.0;

        m2[a][0]=x;
        m2[a][1]=y;
        m2[a][2]=z;
        a2[a][0]=0.0;
        a2[a][1]=0.0;
        a2[a][2]=0.0;
        v2[a][0]=0.0;
        v2[a][1]=0.0;
        v2[a][2]=0.0;
        r[a][0]=0.0;
        r[a][1]=0.0;
        r[a][2]=0.0;

        a=a+1;
    }
}
}
printf("a=%d\n", a);
fflush(stdout);
N=0;
while(N<dm)
{
    x=((double)rand())/32768.0;
    y=((double)rand())/32768.0;
    z=((double)rand())/32768.0;

    if(sqrt((x-0.5)*(x-0.5)+(y-0.5)*(y-0.5)+(z-0.5)*(z-0.5))<=0.2)
    {
        m1[N][0]=x;
        m1[N][1]=y;
        m1[N][2]=z;
        a1[N][0]=0.0;
        a1[N][1]=0.0;
        a1[N][2]=0.0;
        v1[N][0]=0.0;
        v1[N][1]=0.0;
        v1[N][2]=0.0;
        N++;
    }
}

```

```
FILE *output;
```

```

output=fopen("output0.txt","wb");
for(N=0;N<dm;N++){
fwrite(&m1[N][0],sizeof(double), 3, output);
fwrite(&v1[N][0],sizeof(double), 3, output);
}
for(N=0;N<a;N++){
fwrite(&m2[N][0],sizeof(double), 3, output);
fwrite(&v2[N][0],sizeof(double), 3, output);
}

```

```

}
fclose(output);

M=0;
N=0;
R=0;
for(R=1;R<turn+1;R++)
{
    for(N=0;N<dm;N++)
    {
        for(M=N+1;M<dm;M++)
        {
            tempd1=m1[M][0]-m1[N][0];
            tempd2=m1[M][1]-m1[N][1];
            tempd3=m1[M][2]-m1[N][2];
            tempd4=tempd1*tempd1+tempd2*tempd2+tempd3*tempd3;
            tempd5=sqrt(tempd4);
            tempd6=1.0*mas / ((tempd5+r0)*(tempd5+r0));
            tempd1 *= tempd6/(tempd5+r0);
            tempd2 *= tempd6/(tempd5+r0);
            tempd3 *= tempd6/(tempd5+r0);

            a1[N][0]+= tempd1;
            a1[N][1]+= tempd2;
            a1[N][2]+= tempd3;

            a1[M][0]-= tempd1;
            a1[M][1]-= tempd2;
            a1[M][2]-= tempd3;
        }
    }

    for(N=0;N<dm;N++)
    {
        for(M=0;M<a;M++)
        {
            tempd1=m2[M][0]-m1[N][0];
            tempd2=m2[M][1]-m1[N][1];
            tempd3=m2[M][2]-m1[N][2];
            tempd4=tempd1*tempd1+tempd2*tempd2+tempd3*tempd3;
            tempd5=sqrt(tempd4);
            if(tempd5<=0.25)
            {
                tempd6=黑暗能量的密度*mas/ ((tempd5+r0)*(tempd5+r0)) /10.0;
                tempd1 *= tempd6/(tempd5+r0);
                tempd2 *= tempd6/(tempd5+r0);
                tempd3 *= tempd6/(tempd5+r0);

                a1[N][0]+= tempd1;
                a1[N][1]+= tempd2;
                a1[N][2]+= tempd3;

                a2[M][0]-= tempd1;
                a2[M][1]-= tempd2;
                a2[M][2]-= tempd3;
            }
        }
        m1[N][0]+=v1[N][0]*deltaT+0.5*a1[N][0]*deltaT*deltaT;
        m1[N][1]+=v1[N][1]*deltaT+0.5*a1[N][1]*deltaT*deltaT;
    }
}

```

```

m1[N][2]+=v1[N][2]*deltaT+0.5*a1[N][2]*deltaT*deltaT;

v1[N][0]+=a1[N][0]*deltaT;
v1[N][1]+=a1[N][1]*deltaT;
v1[N][2]+=a1[N][2]*deltaT;
}
for(N=0;N<a;N++){
a2[N][0]=-r[N][0]*(8.0*1.20206)*mas/1.0;
a2[N][1]=-r[N][1]*(8.0*1.20206)*mas/1.0;
a2[N][2]=-r[N][2]*(8.0*1.20206)*mas/1.0;

v2[N][0]+=a2[N][0]*deltaT;
v2[N][1]+=a2[N][1]*deltaT;
v2[N][2]+=a2[N][2]*deltaT;

tempd1=v2[N][0]*deltaT+0.5*a2[N][0]*deltaT*deltaT;
tempd2=v2[N][1]*deltaT+0.5*a2[N][1]*deltaT*deltaT;
tempd3=v2[N][2]*deltaT+0.5*a2[N][2]*deltaT*deltaT;

r[N][0]+=tempd1;
r[N][1]+=tempd2;
r[N][2]+=tempd3;

m2[N][0]+=tempd1;
m2[N][1]+=tempd2;
m2[N][2]+=tempd3;
}

if(R%5==0){
strcpy(filename,"output");
strcat(filename,gcvt((float)R,4,buf));
strcat(filename,".txt");
FILE *output1;
output1=fopen(filename,"wb");
for(N=0;N<dm;N++){
fwrite(&m1[N][0],sizeof(double), 3, output1);
fwrite(&v1[N][0],sizeof(double), 3, output1);
}
for(N=0;N<a;N++){
fwrite(&m2[N][0],sizeof(double), 3, output1);
fwrite(&v2[N][0],sizeof(double), 3, output1);
}
fclose(output1);
}

for(N=0;N<dm;N++)
{
a1[N][0]=0.0;
a1[N][1]=0.0;
a1[N][2]=0.0;
}
for(N=0;N<a;N++)
{
a2[N][0]=0.0;
a2[N][1]=0.0;
a2[N][2]=0.0;
}
}
}

```

## 評語

優點：利用簡單的力學原理分析黑暗物質與黑暗能量的斥力與吸力之競爭，可理解宇宙的分布。

缺點：模型乃屬單純，解析之一維結構太理想化，數值結果又樣型偏低可再改進。