臺灣二００八年國際科學展覽會

科　　　　別：數學

作 品 名 稱 ：平面切立方體內單位立方格數極值之計算

得 獎 獎 項 ：佳作

學校 / 作者 ：國立臺南第一高級中學　　　　黃瀚生
國立臺南第一高級中學　　　　洪明道

作者簡介

黃瀚生(右)，就讀於台南一中數理資優班二年級。直到上了國中，數理科才開竅，進而養成對數、理科的高度興趣，並在高中時期，大膽的跨入程式設計的領域。參加過少數競賽(程設、數學)，但因實力與速度俱差而落敗。國中、高一參加的科展比賽成績亦皆不盡理想，直到這次，終於進入台灣區國際科展的複賽，實屬幸運。

洪明道(中)，原居於高雄縣，對科學頗有興趣。後至台南應考資優班，考取後遷到台南。喜好閱讀各類科普叢書、科學雜誌，對數學的研究最初起於幾何；亦廣泛閱讀文學書籍。國中時參加南區科展獲優勝，參加環球盃、青年城市盃分別獲佳作、二等獎，也參與各類私人學辦的數學競賽得獎。高中參加ＴＲＭＬ團體賽獲銀牌。希望藉此次科展能對數學的研究深入一層。

最左邊的是巫昀祐，是我們這個團隊的第三個組員，因國際科展的人數限制而忍痛放棄資格。

作品名稱：平面切立方體內單位立方格數極值之計算

The Calculation of the Maximum Number of the Units that A Plane Cuts A Cube

中文摘要：

我們先假設有一正方體及一截過正方體之平面，並設正立方體爲一 k*k*k 之立體。爲計算平面截過之單位正立方體個數，我們必須先分別計算各層被切過之個數再將之相加，因此將各層面投影至同一平面，簡化爲平面上之問題，並討論其性質／規律，計算平面截此正立方體之個數。如此，便可以一般化數學式計算平面截正立方體個數之問題。

接著，用以上方法爲基礎，討論各種平面切正立方體之類型，將被平面所截之單位立方體個數以電腦程式算出，觀察數字變化及其性質規則，並找出最大値發生之條件。

英文摘要：

We initially supposed that there are a regular hexahedron consists of unitary n×n **cubes and a plane which incises the regular hexahedron. To calculate the total number of the unitary cubes incised by the plane, we can first calculate them layer by layer and then sum them up. And further, we project each layer on the same plane, so the three-dimensional problem is simplified into two-dimension. By making use of the character which results from projection, we can easily calculate the number of the unitary cubes incised. Consequently, we are able to calculate them with a general equation.**

Afterward, we research each circumstance that the plane incises the regular hexahedron on the base of the mentioned methods. Calculate them with self-designed computer programs, and observe the regulation and change of the result. Furthermore, we can find out when it will achieve the maximum.

壹、 前言

一、研究動機

在偶然的機會下，接觸到平面上直線在長方形內部所交之格子數的問題，便希望將其推廣到立體空間中，平面在立方體內相交格子數之計算，並發現立體與平面之對應法則。

二、研究目的

藉由自製之程式計算對給定之平面與給定之立方體，計算所交之格子數。
利用程式將其最佳解解出，並探討其一般性

貳、 研究方法或過程

一、研究器材

(一) Dev C++

(二) Mathematica 4

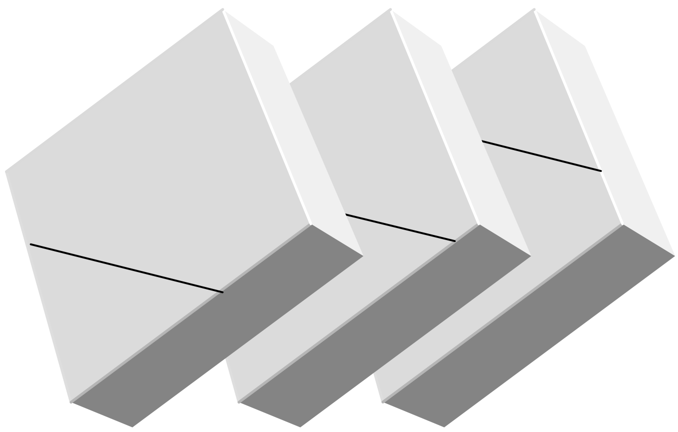
(三) GSP 4.0

二、研究過程或方法

(一)、從正立方體推廣

假設一正立方體分布於第一卦限，其三稜線各於 x ,y ,z 軸上，其邊長爲 n。
若有一平面截立方體，則此截面與 z=0、1、2、….、n 平面之交線必兩兩平行。
|[定義 z=0，1、1，2、2，3、….於正立方體內所夾之空間爲單位立方層。**]|**

如圖：

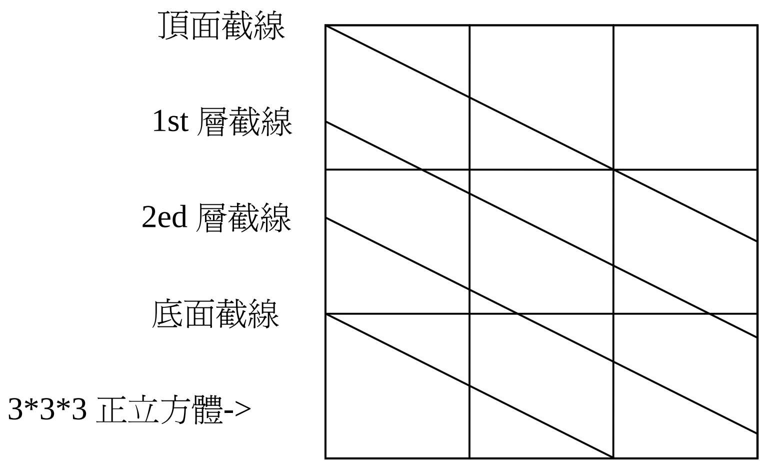


想計算某單位立方格是否與平面相交，只需將其 8 頂點帶入平面之平面方程，檢查是否有出現異號(不含 0)的情況。則欲求立方體內與平面之相交格數只需檢查所有的格子即可。此方法寫成之程式於附錄一，爲一非常沒效率的演算法。

因爲我們將正立方體分成 n 層，所以欲求切面切過之總單位立方體個數，可先算各單位立方層被切過的個數，再將之相加。由於上下平面互相平行，可將這些平

面和截出之兩直線投影於同一平面上，而這些截線在投影面上所包夾的格子數，即爲平面截此單位立方層之個數。將此法用於各單位立方層，即可算出平面截此正立方體之個數。

例：舉 3*3*3 正立方體爲例，應用上述方法投影得下圖



在此投影圖上，頂面截線到 1st 層截線、1st 層截線到底面截線在原立方體中表高爲 1 單位之立方體與平面相交之區塊，故僅考慮頂面截線到 1st 層截線、1st 層截線到底面截線間在投影面上所各自包夾到的格子數，即可求出平面所截出的單位正方體數。上圖之範例爲：頂面截線到 1st 層截線夾 6 格，1st 層截線到 2ed 層截線夾 6 格，2ed 層截線之底(面)截線共夾 5 格，故此平面夾 3*3*3 正立方體 17 格，即此平面共截過 17 個單位立方體。

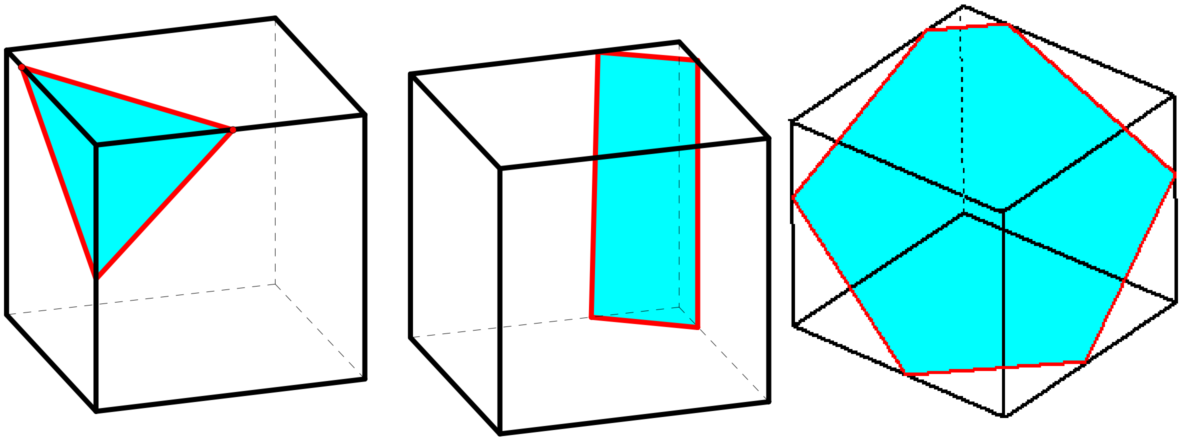
經以上步驟，原爲空間中之問題被簡化成爲平面問題，但又如何以數學式表達被包夾的個數卻又是一個瓶頸。

回顧高中的數列級數題目：平面上 k 條線最多可以將平面切成多少個區塊？其解法爲：假設平面上已有 n-1 條排好的直線，則第 n 條線最多會被 n-1 條線截成 n 段，則每一段可將平面再切出新的一個區塊，共成新的 n 個區塊，n 條線共可成 $\frac{n(n+1)}{2}+1$ 個區塊。同樣的，應用在此題，便可簡便算出被包夾之格子數，在頂面截線到底面截線之層截線被單位方格截成的段數，即爲增加之新區塊數。故若頂截線到底截線夾之方格數爲 m，中間被 k 層截線均分，則可簡單表爲

m+$\sum k$層分成之段數 。

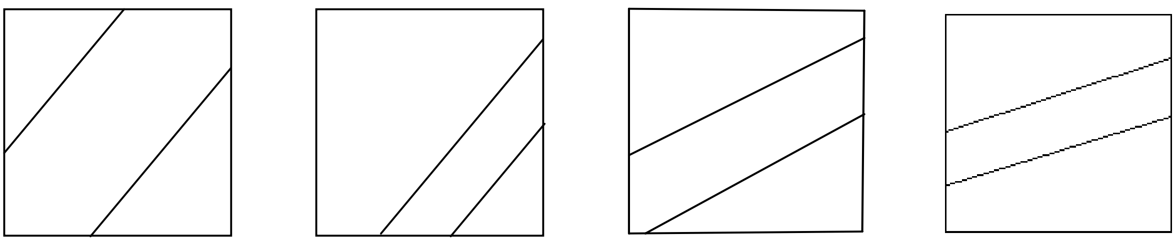
若截線可被一邊長爲 a×b 之最小矩形包覆，則此截線可分成 a+b-(截線所交之格子點數)，特別當截線與矩形之兩頂點相交時，可表成 a+b-gcd(a,b)。比較一般化的計算方法如下：(以斜率爲正者爲例)取該最小矩形之左下角頂點開始，假設其座標爲(x,y)，則將點帶入截線之方程式(x 係數爲正)，若其值爲正，則將 y 值+1；爲負，x 值+1；爲 0，x 或 y 值+1。不斷重複上述步驟，直到 x 或 y 到達邊界爲止，令其最後

之座標爲(u,v)，則段數爲：(u-x)+(v-y)-中途爲 0 之個數-1。又假如此截線與格子點相交，必可找到一極小值之截距變量，使其不與任何格子點相交。



截三面　　　　截四面　　　　截六面

長方體、正立方體與平面相交之情形可分爲截三面、截四面與截六面(單相交於一線或一點皆視爲不相交)，而截五面的情形可視爲平面截一長方體於其頂面或底面上之頂點。討論其投影只需討論以下四種情況：

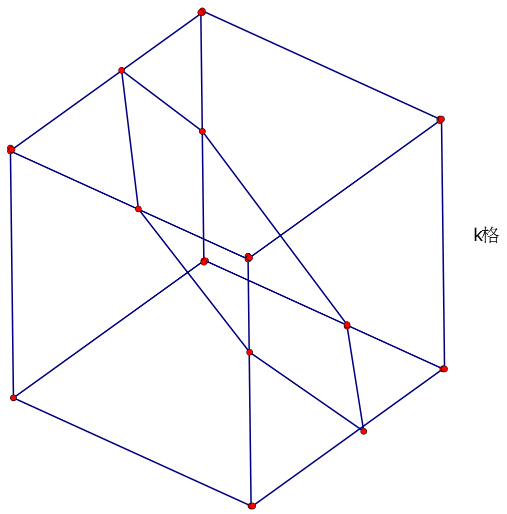


分別是頂截線與底截線交在對角線之同側與異側及一線相交，一線不相交，及兩線皆相交

關於第一個圖，若要計算頂截線與底截線間夾的格子數，則將包含頂、底兩截線的格子及截線以外的格子數自總格子數中扣除(即扣除兩三角型所佔的格子數)，並加上頂、底截線所分成的段數。第二個圖則爲將大三角形所含的格子數扣除小三角型含的格子數加上較小截線的段數。

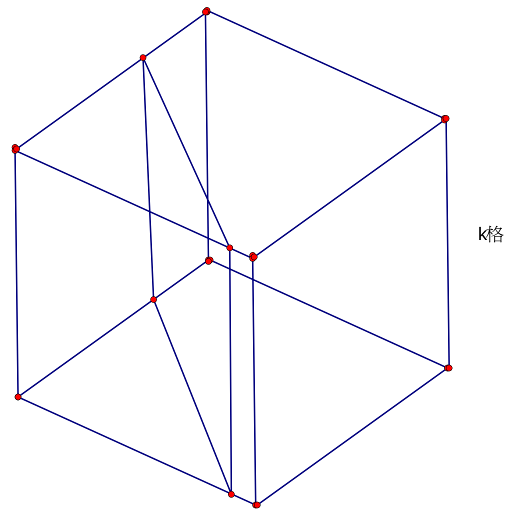
(二)、討論對正立方體所切格子數的極值

關於平面切正立方體的樣式在細分成以下幾種(假設其邊長爲 k)：



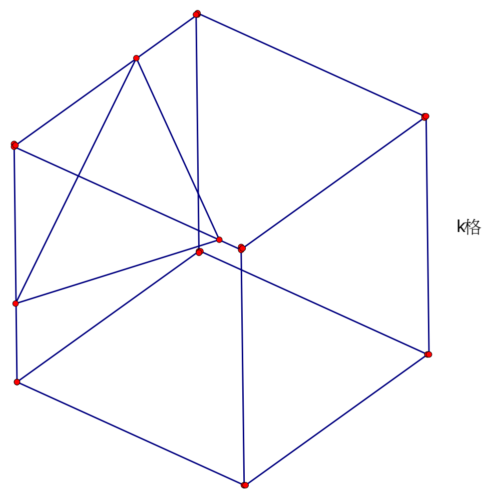
(圖一)

頂、底截線投影至底面分別在對角線異側



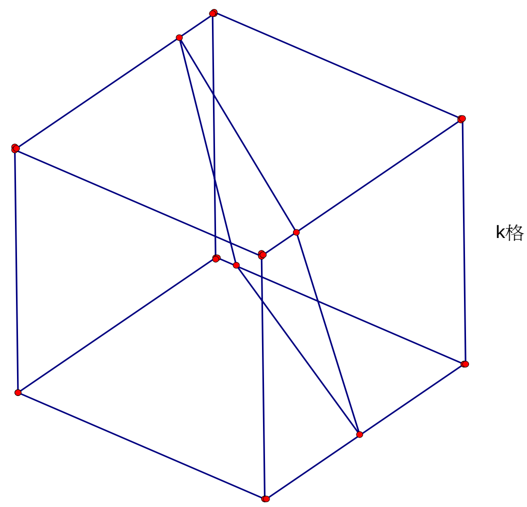
(圖二)

頂、底截線投影至底面分別在對角線同側



(圖三)

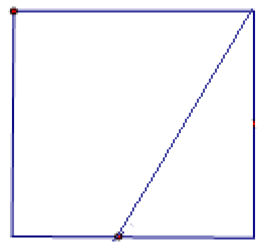
僅含頂截線而未與底面相交



(圖四)

頂截線兩交點交於兩平行邊

其中，圖二的最大值顯而易見可表爲 $(2k-1)k$ ，即頂／底截線交格子數最大值(2k-1)乘上高 k，而圖三之最大值爲三交點在投影方形的頂點之平面，其投影可視爲圖二的一種(同側)，範圍內的格子數其最大值小於等於圖二，在此不需考慮；或其投影可視爲圖一的一種(異側)，則可併入圖一討論。對圖四，必可找到一面作投影，使其變爲如右圖之投影：



可將其與圖一做比較，發現投影與圖一之頂截線退化至正立方體之頂點時

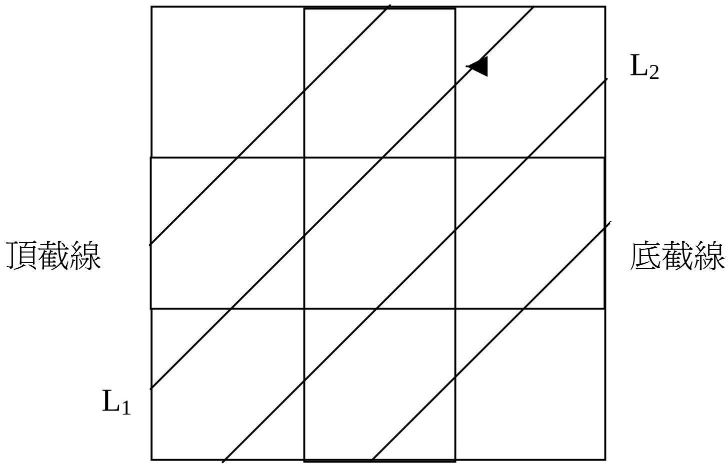
相同，因此將其併入圖一考慮。

對於圖一，在 k=1，其極值為 1，k=2，其極值為 7，k=3，其極值為 19。

k=3 之討論如下：

先考慮截線斜率為 1 之情形

如果是 k 層平面，則共有 k+1 條截線，自頂到底截線依序為 L_1、L_2、L_3…

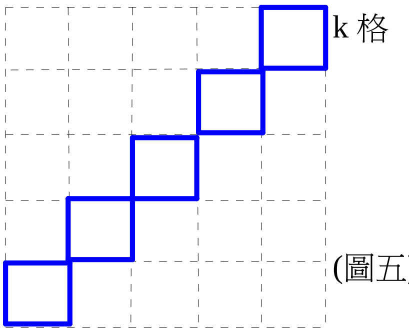


假設 L_2、L_3 所截段數最多，則 L_2 和 L_3 在投影平面上 y 軸之截距小 於 1 大於-1，其差小於 2，故 L_1 y 之截距小於 3，L_4 y 之截距大於－3。又當 L_1 之 y 截距大於 1 且 L_4 之 y 截距小於－1 時中間所夾之方格數為 k^2 為最大值，故當 $1<L_1$ y 截距<3 且 $-1>L_4$ y 截距>-3 時，所夾區塊數(立方格數)有最大值。

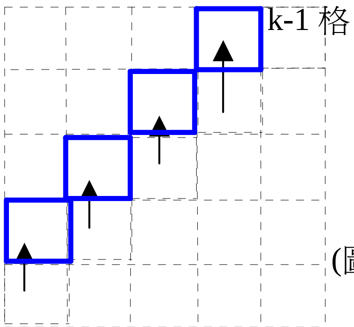
其值為 $k+(k-2)2+(k-3)2+(2k-1)(k-1)=2k^2+2k-5=19$

其中， $2k^2+2k-5$ 所代表的是 $1<L_1$ y 截距<3 且 $-1>L_{k+1}$ y 截距>-3 時夾的格子數 較圖二之極值為大，故不需考慮圖二，僅討論圖一即可。

在討論 k>3 之情況之前，先要將交之格子數之計算代數化，其過程如下：首先，計算頂、底截線之間夾的格子。假設頂、底截線 y 截距 a、－b，且所有截線之 y 截距不為整數。



(圖五)

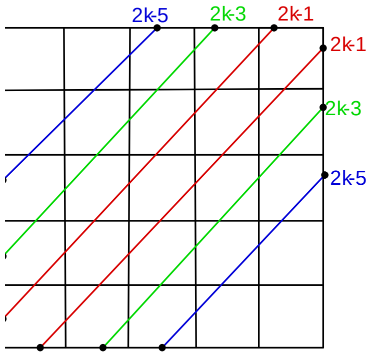


(圖六)

依照之前的討論，則極值存在之平面其頂、底截線間必夾對角線， 其交之格子數為 k，又假若 a>0，則必夾圖四之圖形，較前一排少 1，為 k-1 格。

直到 a+1 所在的那一排，記為 k-([a]+1)格。則累加起來，為 $k+\sum_{i=1}^{i\le[a]+1}(k-[i])$ 格

同理，排除已算過的對角線交的格子，則下半部可累加爲 $\sum_{i=1}^{i\le[b]+1}(k-[i])$ 格



假設頂、底截線間之截線皆不與格子點相交。
從被截段數最多者開始依序遞減，爲 2k-1、2k-3、2k-5……，爲一等差數列。

故其所截段數總和可表爲：

$$(2k-2\times[\left|a-\frac{a+b}{k}\right|]-1)+(2k-2\times[\left|a-\frac{a+b}{k}\times 2\right|]-1)$$

$$+(2k-2\times[\left|a-\frac{a+b}{k}\times 3\right|]-1)+.....$$

$$+(2k-2\times[\left|a-\frac{a+b}{k}\times(k-1)\right|]-1)=\sum_{i=1}^{i<k}(2k-2\times[\left|a-\frac{a+b}{k}i\right|]-1)$$

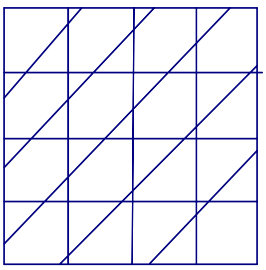
則全部加總，得 $k+\sum_{i=1}^{i\le[a]+1}(k-i)+\sum_{i=1}^{i\le[b]+1}(k-i)+\sum_{i=1}^{i<k}(2k-2\times[\left|a-\frac{a+b}{k}i\right|]-1)$

此即爲頂、底截線 y 截距 a、-b，邊長 k 所交之格子數
假設 k 爲已知數，則將其改寫成：

$$k+(k-1)\times(2k-1)+\sum_{i=1}^{i\le[a]+1}(k-i)+\sum_{i=1}^{i\le[b]+1}(k-i)-2\times\sum_{i=1}^{i<k}[\left|a-\frac{a+b}{k}i\right|]$$

$$=2k^2+k([a]+[b])+1-\frac{[a]^2+[b]^2+3([a]+[b])+4}{2}-2\sum_{i=1}^{i<k}[\left|a-\frac{a+b}{k}i\right|]......式(1)$$

k=4



由 k=3 之情形類推，可知極值所在之頂、底截線 y 軸截距必大於 1、小於-1。

將 k=4 帶入式(1)，得

$$33+4([a]+[b])-\frac{[a]^2+[b]^2+3([a]+[b])+4}{2}-2\sum_{i=1}^{i<4}[\left|a-\frac{a+b}{4}i\right|]$$

假設$a > b$，而$[a] = 1$且$[b] = 1$，則得$36 - 6 - 2(....) \le 30$
若$[a] = 2$且$[b] = 1$，則得$36 - 2(1 + ...) \le 34$
若$[a] = 2$且$[b] = 2$，則得$37 - 2(1 + ...) \le 35$
剩下的可以不需討論，因爲由圖中可知當a、b皆大於2，
即可包含所有格子，向外只會讓中間之截線被截之段數減少。

又假如能找到一組a、b使其值爲35，
即可確定35爲最大值。
取$2 \le a < 3$，$2 \le b < 3$
及$8 > 3a - b > 4$ 、$-a + 3b < 4$取交集
$3b < 4 + a$，又$a < 3$，得$2 < b < 7/3$
代回$3a > 4 + b$，得$19/9 < a < 3$，且$a > b$
代回原式，求得35，即爲其最大值

其他之情形以程式跑第(1)式求其極值較爲簡便
以下是以附錄二之程式跑出的結果(以頂、底截線 y 軸截距 k、－k，斜率 1，間隔 0.021 解出之結果)

k	交格數(max)	L_1 y 截距	L_{k+1} 截距
1	1	1	-1
2	7	2	-1.979
3	19	2.979	-2.979
4	35	3.475	-2.488
5	57	3.32	-3.32
6	82	4.236	-3.228
7	113	4.186	-4.186
8	147	5.144	-4.157
9	187	5.136	-5.136
10	230	6.115	-5.107
11	279	6.107	-6.107
12	331	7.086	-6.078
13	389	7.078	-7.057
14	450	8.078	-7.07
15	517	8.07	-8.07
16	587	9.07	-8.062
17	663	9.062	-9.062
18	742	10.041	-9.054
19	827	10.054	-10.054
20	915	11.054	-10.046

以上之 y 截距爲第一個解出之極大值，觀察上表，可發現：

L1 y 截距呈 1, 2, 2, 3, 3, 4, 4, 5, 5, 6, 6, 7, 7, 8, 8, 9, 9, 10, 10, 11, 11, 12, 12

Lk+1 y 截距呈-1, -1, -2, -2, -3, -3, -4, -4, -5, -5, -6, -6, -7, -7, -8, -8, -9, -9, -10, -10, -11, -11 之數列成長，僅在小數部份不同，且 k 越大，y 截距可形成極值的範圍越小。此一猜測似乎對於縮小搜尋範圍極有效。依此猜測所建之程式在附錄三。除此之外，還發現最大值有一個美妙的規律(見左方)：

1	6	6
7	12	4
19	16	6
35	22	3
57	25	6
82	31	3
113	34	6
147	40	3
187	43	6
230	49	3
279	52	6
331	58	3
389	61	6
450	67	3
517	70	6
587	76	3
663	79	6
742	85	3
827	88	
915		

↑規律

令第二層階差數列自第 3 項起爲數列 g(i)(i 自第三項算起)，原數列自第三項開始爲 f(k-2)

令 g(i)自第一項起爲 4，7，13，16，22，25，31，34，40……

第 i　0　1　2　3　4　5　6　7　8

建構一數列 0，1，1，2，2，3，3，4，4……

可得其第 i 項爲 $\frac{2i+3}{4}+\frac{(-1)^i}{4}$

得 g(i)= $\frac{11-3(-1)^i}{2}+9\left(\frac{2i+3}{4}+\frac{(-1)^i}{4}\right)$

又 f(k-2)=f(k-1-2)+g(k-1)，f(1)=19

$$\Rightarrow \mathrm{f}(\mathrm{k}-2)=\mathrm{f}(1)+\sum_{\mathrm{j}=3}^{\mathrm{k}-1}\mathrm{g}(\mathrm{j})$$

$$\therefore \mathrm{f}(\mathrm{k}-2)=19+\frac{11}{2}(k-3)-\frac{3}{2}(-1+1-1+1.....+(-1)^{k-3})+$$

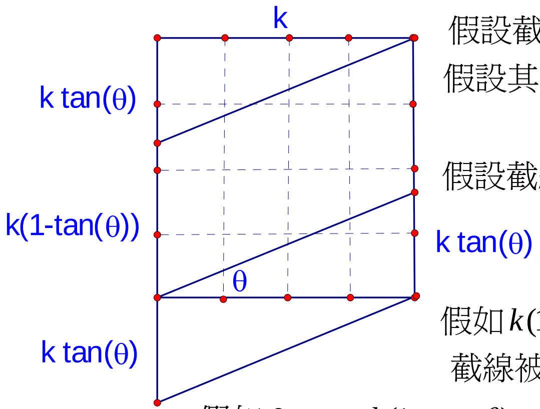
$$\frac{9}{4}\sum_{i=1}^{k-3}(2\mathrm{i}+3)+\frac{9}{4}(-1+1-1...(-1)^{k-3})$$

$$=\frac{5}{2}+\frac{11}{2}\mathrm{k}+\left(\frac{3}{4}\right)\left(\frac{(-1)^{\mathrm{k}-3}-1}{2}\right)+\frac{9}{4}(\mathrm{k}+1)(\mathrm{k}-3)$$

$$\text{則 } \mathrm{m(k)} = \begin{cases} \text{if } \mathrm{k}=1 \quad \mathrm{m(k)}=1 \\ \text{else if } \mathrm{k}=2 \quad \mathrm{m(k)}=7 \\ \text{else } \mathrm{m(k)}=\frac{5}{2}+\frac{11}{2}\mathrm{k}+\left(\frac{3}{4}\right)\left(\frac{(-1)^{\mathrm{k}-3}-1}{2}\right)+\frac{9}{4}(\mathrm{k}+1)(\mathrm{k}-3) \end{cases}$$

此概念寫成之程式放在附錄四，在 k<50000 都已獲得証實合於最大值。其時間維度爲 O(1)。

(四)、對斜率不為 1 之截線討論



假設截線與一邊夾 θ，則令 $0<\theta<\pi/4$ 以避免重複討論。

假設其中一條截線在 y 軸截距為 y_0。

假設截線未與格子點相交，則可以將其分成三區段：截線 y 軸截距 $k(1-\tan\theta)<y_0<k$、$0<y_0\le k(1-\tan\theta)$、$-k\tan\theta<y_0<0$。

假如 $k(1-\tan\theta)<y_0<k$

截線被截段數為 $\lceil k-y_0\rceil+\lceil (k-y_0)\cot(\theta)\rceil-1$

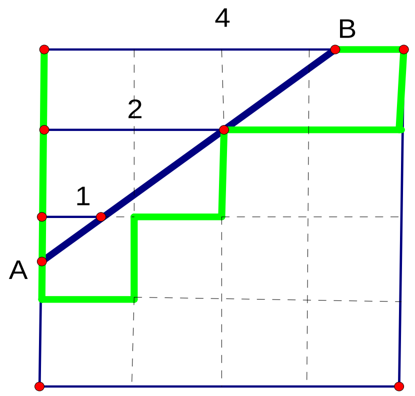
假如 $0<y_0\le k(1-\tan\theta)$

截線被截段數為 $\lceil y_0+k\ \tan(\theta)\rceil-[y_0]+k-1$

假如 $-k\tan\theta<y_0<0$

截線被截段數為 $\lceil (k\tan(\theta)+y_0)\cot(\theta)\rceil+\lceil k\tan(\theta)+y_0\rceil-1$

又假如頂底截線與 y 軸之截距為 a、-b，a>0，-b<0，則在 $a>k(1-\tan(\theta))$ 才有可能有最大值之存在。因此討論 $a>k(1-\tan(\theta))$，-b<0 時之情況



截線與相鄰兩軸所圍的區域內之格子數(包含不完整之格子)：

設一直線 AB，則自 OA 向上算，每一水平線至截線的長度(取 Ceiling 符號)即為該線段所對應的格子數

因此將其加總，可得：1+2+4=7

O　　因此兩截線內夾的完整格子數為

$$k^2-\sum_{i=[a]+1}^{i\le k}\left\lceil \frac{i-a}{\tan(\theta)}\right\rceil-\sum_{i=0}^{i\le[k\tan(\theta)-b]}\left\lceil k-\frac{i+b}{\tan(\theta)}\right\rceil$$

註：此式有可能小於 0，其意義請看**肆、討論及應用 五**

此平面截之格子數即為

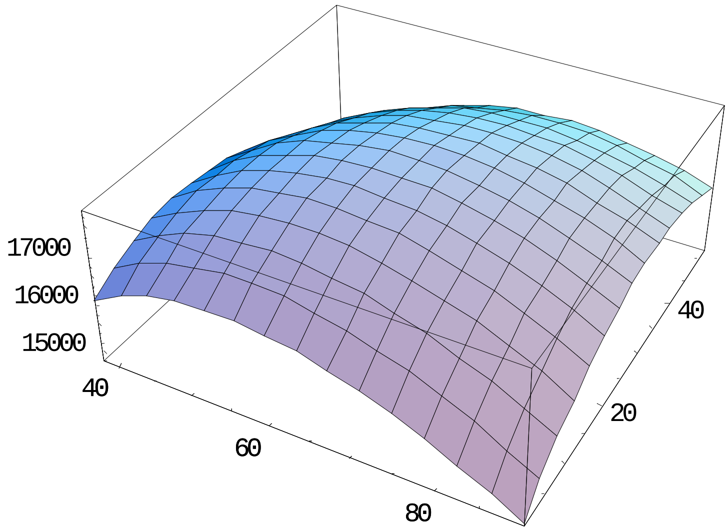
$$k^2-\sum_{i=[a]+1}^{i\le k}\left\lceil \frac{i-a}{\tan\theta}\right\rceil-\sum_{i=0}^{i\le[k\tan(\theta)-b]}\left\lceil k-\frac{i+b}{\tan\theta}\right\rceil+$$

$$\sum_{i=\left\lceil\frac{k(a-k(1-\tan\theta))}{a+b}\right\rceil}^{i<\frac{ka}{a+b}}(\left\lceil a+\frac{a+b}{k}i+k\tan\theta\right\rceil-[a+\frac{a+b}{k}i])+k-1)$$

$$\sum_{i=0}^{i<\frac{k(a-k(1-\tan\theta))}{a+b}}(\left\lceil k-a+\frac{a+b}{k}i\right\rceil+\left\lceil (k-a+\frac{a+b}{k}i)\cot\theta\right\rceil-1)$$

$$+\sum_{i=\left\lceil\frac{ka}{a+b}\right\rceil}^{i\le k}(\left\lceil (k\tan\theta+a-\frac{a+b}{k}i)\cot\theta\right\rceil+\left\lceil k\tan\theta+a-\frac{a+b}{k}i\right\rceil-1)\text{.........式(2)}$$

下圖爲 k=90，$\theta = 30°$，格子數爲 z 軸，a、b 爲 x、y 軸所繪出之圖形，可與附錄五之圖比較。



(五)、投影之直線轉換回平面

假設 L_1：$ax+by+c_1=0$、

L_{k+1}：$ax+by+c_2=0$ 爲所要轉換之頂、底截線　方程式

且 b=0 或 $\frac{-a}{b} \geq 0$，則令 y=0 解出 L1：$x=-c_1/a$，Lk+1：$x=-c_2/a$，再令 x=0 解出 L_{k+1}：$y=-c_2/b$，則解出之三點在空間中爲：$(-c_1/a, 0, k)$、$(-c_2/a, 0, 0)$、$(0, -c_2/b, 0)$

平面法向量：$(\frac{k}{b}, \frac{k}{a}, \frac{c_1-c_2}{ab})$

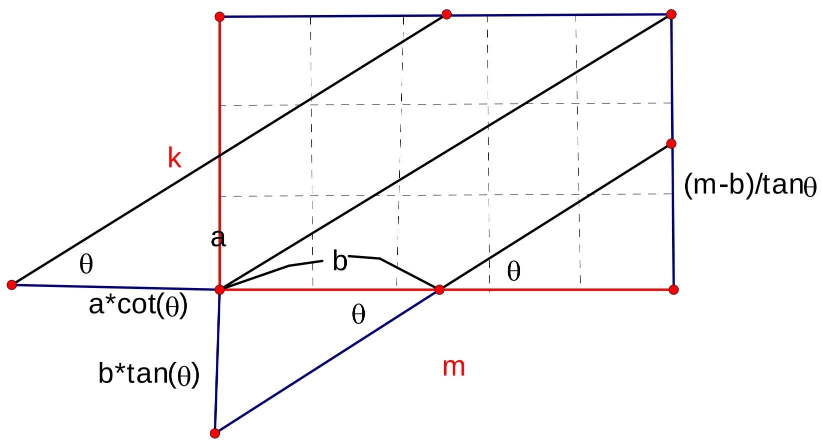
點帶入，得 $\frac{k}{b}x+\frac{k}{a}y+\frac{c_1-c_2}{ab}z=-\frac{k \times c_2}{ab}$

(六)、推廣=>矩形體

接下來要討論的是長、寬、高長度爲 m、k、l 的矩形體。(m>k，$m,k,l \in N$)

以長、寬所在的平面爲投影面(x-y 平面)，令 k 對應 y 軸，m 對應 x 軸。

假設所求之平面投影於 x-y 平面交 y 軸於 a，x 軸於 b，並令頂截線爲 $y-a=x\tan\theta$，底截線 $y=\tan\theta(x-b)$頂、底截線投影與 x 軸夾 θ 角。



總格子數爲 k×m

頂截線以上所包夾的格子數：$\sum_{i=[a]+1}^{i\le k}\left\lceil \frac{i-a}{\tan\theta} \right\rceil$

底截線以外所包夾的格子數：$\sum_{i=[b]+1}^{i\le m}\left\lceil (i-b)\tan\theta \right\rceil$

頂、底截線包夾區域的完整格子數：

$$\text{k×m}-\sum_{i=[a]+1}^{i\le k}\left\lceil \frac{i-a}{\tan\theta} \right\rceil-\sum_{i=[b]+1}^{i\le m}\left\lceil (i-b)\tan\theta \right\rceil \quad \cdots\cdots \text{式(3)}$$

註：此式有可能小於 0，其意義請看**肆、討論及應用 五**

1.假設 $\tan\theta$ =k/m

將其帶入式(3)，得 $k\times m-\sum_{i=[a]+1}^{i\le k}\left\lceil \frac{m(i-a)}{k} \right\rceil-\sum_{i=[b]+1}^{i\le m}\left\lceil \frac{k(i-b)}{m} \right\rceil$

對角線左上之截線所截段數總和：

$$\sum_{i=0}^{i\le \frac{al}{a+bk/m}}\left(\left\lceil k-a+\frac{a+\frac{bk}{m}}{l}\times i \right\rceil+\left\lceil (k-a+\frac{a+\frac{bk}{m}}{l}\times i)\frac{m}{k} \right\rceil-1\right)$$

對角線右下之截線所截段數總和：

$$\sum_{i=0}^{\frac{bl}{am/k+b}>i}\left(\left\lceil m-b+\frac{\frac{am}{k}+b}{l}\times i \right\rceil+\left\lceil \frac{(m-b)k}{m}+\frac{a+\frac{bk}{m}}{l}\times i \right\rceil-1\right)$$

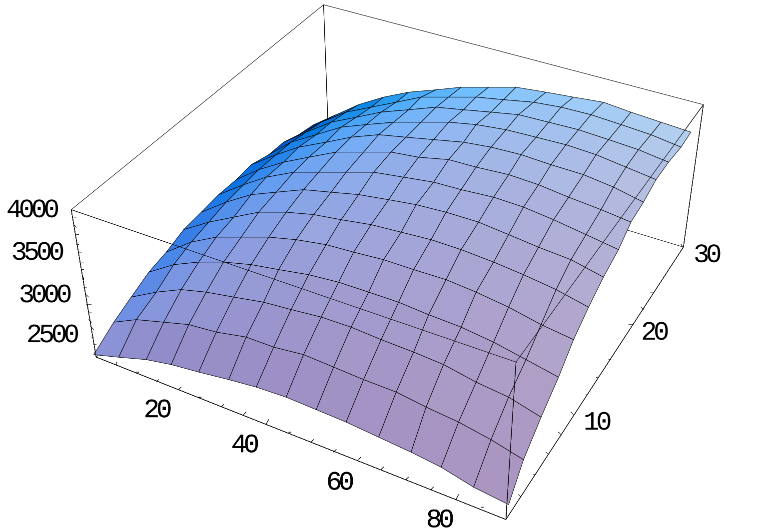
故可得總格數：

$$k\times m-\sum_{i=[a]+1}^{i\le k}\left\lceil \frac{m(i-a)}{k} \right\rceil-\sum_{i=[b]+1}^{i\le m}\left\lceil \frac{k(i-b)}{m} \right\rceil+$$

$$\sum_{i=0}^{i\le \frac{al}{a+bk/m}}\left(\left\lceil k-a+\frac{a+\frac{bk}{m}}{l}\times i \right\rceil+\left\lceil (k-a+\frac{a+\frac{bk}{m}}{l}\times i)\frac{m}{k} \right\rceil-1\right)$$

$$+\sum_{i=0}^{\frac{bl}{am/k+b}>i}\left(\left\lceil m-b+\frac{\frac{am}{k}+b}{l}\times i \right\rceil+\left\lceil \frac{(m-b)k}{m}+\frac{a+\frac{bk}{m}}{l}\times i \right\rceil-1\right)$$

(k=90,m=30,l=20 時所繪之圖)



2. θ =0

平面所截之格數最大值爲 k×m+l×m

3. **k/m<tan θ < k/ (m-b)**

α 區：x 軸截距大於 0 小於 m-k/tan θ

截線被截段數：

$$k+\lceil x+k/\tan\theta\rceil-[x]-1$$

β 區：y 軸截距大於 0 小於 a

截線被截段數：

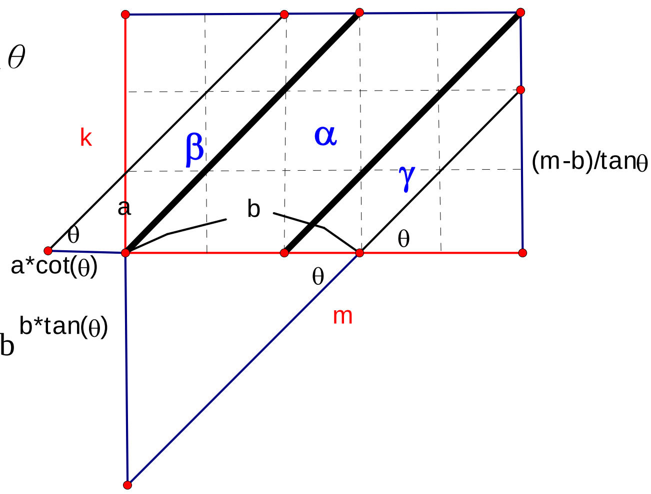
$$\lceil k-y\rceil+\lceil (k-y)/\tan\theta\rceil-1$$

γ 區：x 軸截距大於 m-k/tan θ 小於 b

截線被截段數：

$$\lceil m-x\rceil+\lceil (m-x)\tan\theta\rceil-1$$

平面所截之格數：



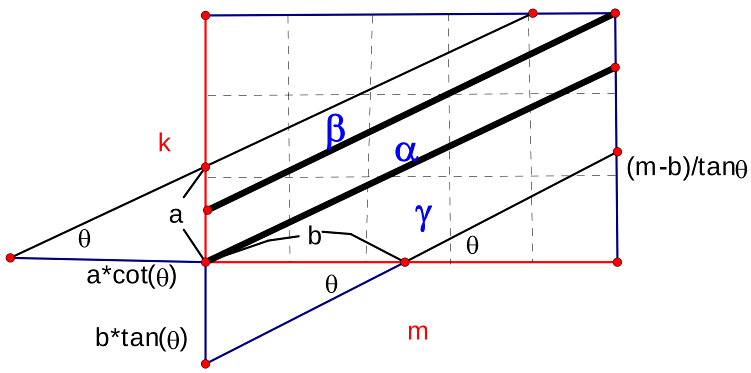
$$\mathrm{k}\times\mathrm{m}-\sum_{i=[a]+1}^{i\le k}\left\lceil\frac{i-a}{\tan\theta}\right\rceil-\sum_{i=[b]+1}^{i\le m}\lceil (i-b)\tan\theta\rceil+$$

$$\sum_{i=0}^{i\le\frac{al}{a+b\times\tan\theta}}\left(\left\lceil k-a+i\times\frac{a+b\cdot\tan\theta}{l}\right\rceil+\left\lceil\left(k-a+i\times\frac{a+b\cdot\tan\theta}{l}\right)\cot\theta\right\rceil-1\right)$$

$$+\sum_{i=\left\lceil\frac{-m+k/\tan\theta+b}{b+a\cot\theta}l\right\rceil}^{i<\frac{bl}{b+a\cot\theta}}\left(k+\left\lceil b-\frac{b+a\cot\theta}{l}i+k/\tan\theta\right\rceil-\left[b-\frac{b+a\cot\theta}{l}i\right]-1\right)+$$

$$\sum_{i=0}^{i<\frac{l(b-m+k/\tan\theta)}{a\times\cot\theta+b}}\left(\left\lceil m-b+i\times\frac{a\times\cot\theta+b}{l}\right\rceil+\left\lceil\left(m-b+i\times\frac{a\times\cot\theta+b}{l}\right)\tan\theta\right\rceil-1\right)$$

4. **k/m>tan θ > (k-a) /m**

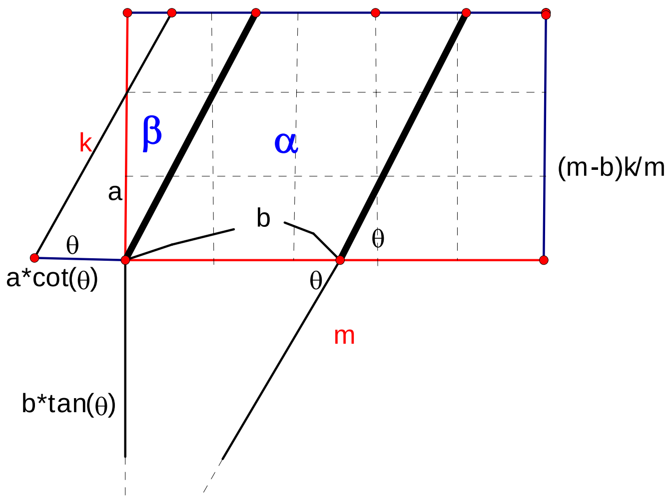


即將 3.之結果代換成：

$a\leftrightarrow b$，$\cot\theta\leftrightarrow\tan\theta$，

$k\leftrightarrow m$

5. **tan θ > k/ (m-b)**



所夾完整格子數：$k\times m-\sum_{i=[a]+1}^{i\le k}\left\lceil\frac{i-a}{\tan\theta}\right\rceil-\sum_{i=[b]+1}^{[k\times\cot\theta+b]}\lceil(i-b)\tan\theta\rceil-\sum_{i=[k\times\cot\theta+b]+1}^{m}k$

平面總截格子數：

$$k\times m-\sum_{i=[a]+1}^{i\le k}\left\lceil\frac{i-a}{\tan\theta}\right\rceil-\sum_{i=[b]+1}^{[k\times\cot\theta+b]}\lceil(i-b)\tan\theta\rceil-\sum_{i=[k\times\cot\theta+b]+1}^{m}k$$

$$+\sum_{i=0}^{i\le\frac{al}{a+b\times\tan\theta}}\left(\left\lceil k-a+i\times\frac{a+b\cdot\tan\theta}{l}\right\rceil+\left\lceil\left(k-a+i\times\frac{a+b\cdot\tan\theta}{l}\right)\cot\theta\right\rceil-1\right)$$

$$+\sum_{i=0}^{i<\frac{bl}{a\times\cot\theta+b}}\left(k+\left\lceil b-\frac{b+a\cot\theta}{l}i+k/\tan\theta\right\rceil-\left[b-\frac{b+a\cot\theta}{l}i\right]-1\right)$$

6. tan θ <(k-a) /m

即將 5.之結果代換成：$a\leftrightarrow b$，$\cot\theta\leftrightarrow\tan\theta$，$k\leftrightarrow m$

參、 結論

一、若知道平面與正立方體之截線，則可將其投影於一面，並計算出頂底截線間包夾之區塊數(式(1))。

二、假若圖形未與格子點相交，且截線與邊夾 45°時，可套用

$$2k^2+k([a]+[b])+1-\frac{[a]^2+[b]^2+3([a]+[b])+4}{2}-2\sum_{i=1}^{i<k}\left[\left|a-\frac{a+b}{k}i\right|\right]\ldots\text{式}(1)$$

以求其交格數。

三、式(1)之最大值似乎都發生在截線與邊夾 45°時

四、假如 k<50000，要求截線與邊夾 45°時(式(1))之最大值可帶

$$m(k)=\begin{cases}\text{if } k=1 \quad m(k)=1\\ \text{else if } k=2 \quad m(k)=7\\ \text{else } m(k)=\frac{5}{2}+\frac{11}{2}k+\left(\frac{3}{4}\right)\left(\frac{(-1)^{k-3}-1}{2}\right)+\frac{9}{4}(k+1)(k-3)\end{cases}$$

五、長方體被平面所截之格數關係圖係由多條不同之方程式所決定。

六、當立方體邊長極大時，以 a 為 x 軸，b 為 y 軸，式(1)或式(2)為 z 軸所繪之圖巨觀下趨近於一平滑曲面，但縮小比例尺，仍可發覺其圖形之不連續性。

七、以程式找出長方體之極值(長寬高分別為 k, m, l)：

2 3 6 (k, m, l)

All：26 格 a=0.650 b=0.010 tan(θ)=0.360

4 6 12 (k, m, l)

All：115 格 a=0.990 b=0.090 tan(θ)=0.510

6 9 18 (k, m, l)

All：264 格 a=1.050 b=0.630 tan(θ)=0.591

8 12 24 (k, m, l)

All：473 格 a=0.930 b=0.850 tan(θ)=0.635

10 15 30 (k, m, l)

All：742 格 a=1.030 b=0.850 tan(θ)=0.635

(tan(θ)指的是從θ=0~pi/2 間發生極值之最小角所對應之 tan，θ間格為 0.01)

猜測期極值可能在 tan(θ)=k/m 時發生。

肆、 討論及應用

一、類似的方法可以再用於長方體、錐體之討論。較一般化之做法如一開始給之計算方法，或附錄一之程式的演算概念(對每一格子的各個頂點帶入平面方程式，並檢查其正負以判斷此格子是否有與平面相交)。

二、關於為何截線之斜率為 1 時必有最大值存在及上述性質的原因仍不明。

程式跑出來的極大值似乎都偏向斜率 1 的。

三、我們所得知求交格子個數之公式(1)尚無法證出其最大值之通式(據猜測通式為 m(k))，以 Mathematica 似乎也莫可奈何。目前僅能對給定之 k 分別進行討論以求其極值。

四、關於式(1)極值所做的努力：

首先，將造成函數不連續的因子去除(高斯符號)，方法如下：

建立一函數，s(x)=x，以 0~1 為週期。

代入傅立葉級數定義：f(t)= $a_0+\sum_{n=1}^{\infty} a_n\cos\left(\frac{2\pi}{b-a}nt\right)+b_n\sin\left(\frac{2\pi}{b-a}nt\right)$

得 $\mathrm{Gauss}(x)=\frac{1}{2}+\frac{i\ln\left(1-e^{-2i\pi t}\right)}{2\pi}-\frac{i\ln\left(1-e^{2i\pi t}\right)}{2\pi}$

令 $g(a,b)=2k^2+k\left(\mathrm{Gauss}(a)+\mathrm{Gauss}(b)\right)+1-2\sum_{i=1}^{k-1}\mathrm{Gauss}\left(\sqrt{(a-(a+b)/k*i)^2}\right)-\frac{1}{2}\left((\mathrm{Gauss}(a))^2+(\mathrm{Gauss}(b))^2+3\left(\mathrm{Gauss}(a)+\mathrm{Gauss}(b)\right)+4\right)$

則對給定之常數 k，即一定範圍之 a、b，-g(a, b)可以在 Mathematica 上做 FindMinimun 的動作，以求 g(a, b)之極值。但似乎仍無法解出通解。

五、關於式(3)小於 0 之值，會在計算頂、底截線被截之段數後加回，加回後之值即為頂、底截線間包夾之(完整+不完整)格數。

伍、 參考文獻

一、阿壽工坊 http://140.111.115.8/longlife/

二、初等線性代數 作者：林義雄

三、投影幾何 作者：魏義峰

四、投影理論基礎 作者：徐宏文,郭紹仲

五、投影幾何學(修訂版) 作者：王照明 出版社：全華科技

六、數學傳播第 22 卷 第 3 期

七、從尤拉公式到空間的平面分割截面的相關體元搜索演算法及應用

作者：王威信[1] 王德信[2][1]瀋陽工業學院機械設計系 [2]吉林醫學院附屬醫院

八、數學運算大師 MATHEMATICA4 作者：洪維恩

陸、　　附錄

一、最初用以檢驗之程式

時間維度為 $O(k^6)$

```
#include<iostream>
using namespace std;
main()
{
   int m,n,o,na=0;
   double x,y,z,xmax,ymax,zmax;
   cin>>m>>n>>o;
    for (x=0;x<m;x+=0.1)
    {
      for (y=0;y<n;y+=0.1)
      {
         for (z=0;z<o;z+=0.1)
         {
            bool t[100][100][100]={0};
            int i,j,k,c=0;
            for (i=0;i<m;i++)
            for (j=0;j<n;j++)
            for(k=0;k<o;k++)
            {
               if (i/x+j/y+k/z-1>0)
               {
                  if ((i+1)/x+j/y+k/z-1<0)t[i][j][k]=1;
                  else if (i/x+(j+1)/y+k/z-1<0)t[i][j][k]=1;
                  else if (i/x+j/y+(k+1)/z-1<0)t[i][j][k]=1;
                  else if ((i+1)/x+(j+1)/y+k/z-1<0)t[i][j][k]=1;
                  else if ((i+1)/x+j/y+(k+1)/z-1<0)t[i][j][k]=1;
                  else if (i/x+(j+1)/y+(k+1)/z-1<0)t[i][j][k]=1;
                  else if ((i+1)/x+(j+1)/y+(k+1)/z-1<0)t[i][j][k]=1;
               }
               else if (i/x+j/y+k/z-1<0)
               {
               else if ((i+1)/x+j/y+k/z-1>0)t[i][j][k]=1;
               else if (i/x+(j+1)/y+k/z-1>0)t[i][j][k]=1;
               else if (i/x+j/y+(k+1)/z-1>0)t[i][j][k]=1;
               else if ((i+1)/x+(j+1)/y+k/z-1>0)t[i][j][k]=1;
```

```
else if ((i+1)/x+j/y+(k+1)/z-1>0)t[i][j][k]=1;
else if (i/x+(j+1)/y+(k+1)/z-1>0)t[i][j][k]=1;
else if ((i+1)/x+(j+1)/y+(k+1)/z-1>0)t[i][j][k]=1;
}
else if (i/x+(j+1)/y+k/z-1>0)
{
if (i/x+j/y+k/z-1<0)t[i][j][k]=1;
else if ((i+1)/x+j/y+k/z-1<0)t[i][j][k]=1;
else if (i/x+j/y+(k+1)/z-1<0)t[i][j][k]=1;
else if ((i+1)/x+(j+1)/y+k/z-1<0)t[i][j][k]=1;
else if ((i+1)/x+j/y+(k+1)/z-1<0)t[i][j][k]=1;
else if (i/x+(j+1)/y+(k+1)/z-1<0)t[i][j][k]=1;
else if ((i+1)/x+(j+1)/y+(k+1)/z-1<0)t[i][j][k]=1;
}
else if (i/x+(j+1)/y+k/z-1<0)
{
if (i/x+j/y+k/z-1>0)t[i][j][k]=1;
else if ((i+1)/x+j/y+k/z-1>0)t[i][j][k]=1;
else if (i/x+j/y+(k+1)/z-1>0)t[i][j][k]=1;
else if ((i+1)/x+(j+1)/y+k/z-1>0)t[i][j][k]=1;
else if ((i+1)/x+j/y+(k+1)/z-1>0)t[i][j][k]=1;
else if (i/x+(j+1)/y+(k+1)/z-1>0)t[i][j][k]=1;
else if ((i+1)/x+(j+1)/y+(k+1)/z-1>0)t[i][j][k]=1;
}
else if (i/x+j/y+(k+1)/z-1>0)
{
if (i/x+j/y+k/z-1<0)t[i][j][k]=1;
else if ((i+1)/x+j/y+k/z-1<0)t[i][j][k]=1;
else if (i/x+(j+1)/y+k/z-1<0)t[i][j][k]=1;
else if ((i+1)/x+(j+1)/y+k/z-1<0)t[i][j][k]=1;
else if ((i+1)/x+j/y+(k+1)/z-1<0)t[i][j][k]=1;
else if (i/x+(j+1)/y+(k+1)/z-1<0)t[i][j][k]=1;
else if ((i+1)/x+(j+1)/y+(k+1)/z-1<0)t[i][j][k]=1;
}
else if(i/x+j/y+(k+1)/z-1<0)
{
if (i/x+j/y+k/z-1>0)t[i][j][k]=1;
else if ((i+1)/x+j/y+k/z-1>0)t[i][j][k]=1;
else if (i/x+(j+1)/y+k/z-1>0)t[i][j][k]=1;
else if ((i+1)/x+(j+1)/y+k/z-1>0)t[i][j][k]=1;
```

```
                else if ((i+1)/x+j/y+(k+1)/z-1>0)t[i][j][k]=1;
                else if (i/x+(j+1)/y+(k+1)/z-1>0)t[i][j][k]=1;
                else if ((i+1)/x+(j+1)/y+(k+1)/z-1>0)t[i][j][k]=1;
                }
                else if((i+1)/x+(j+1)/y+k/z-1>0)
                {
                if (i/x+j/y+k/z-1<0)t[i][j][k]=1;
                else if((i+1)/x+j/y+k/z-1<0)t[i][j][k]=1;
                else if(i/x+(j+1)/y+k/z-1<0)t[i][j][k]=1;
                else if(i/x+j/y+(k+1)/z-1<0)t[i][j][k]=1;
                else if((i+1)/x+j/y+(k+1)/z-1<0)t[i][j][k]=1;
                else if(i/x+(j+1)/y+(k+1)/z-1<0)t[i][j][k]=1;
                else if((i+1)/x+(j+1)/y+(k+1)/z-1<0)t[i][j][k]=1;
                }
                else if((i+1)/x+(j+1)/y+k/z-1<0)
                {
                if (i/x+j/y+k/z-1>0)t[i][j][k]=1;
                else if((i+1)/x+j/y+k/z-1>0)t[i][j][k]=1;
                else if(i/x+(j+1)/y+k/z-1>0)t[i][j][k]=1;
                else if(i/x+j/y+(k+1)/z-1>0)t[i][j][k]=1;
                else if((i+1)/x+j/y+(k+1)/z-1>0)t[i][j][k]=1;
                else if(i/x+(j+1)/y+(k+1)/z-1>0)t[i][j][k]=1;
                else if((i+1)/x+(j+1)/y+(k+1)/z-1>0)t[i][j][k]=1;
              }
              if(t[i][j][k])c++;
            }
            if(c>na){na=c;ymax=y;zmax=z;xmax=x;cout<<na<<"\n";}
            }
        }
    }
    cout<<xmax<<" "<<ymax<<" "<<zmax<<"\n"<<na<<"\n";
    system("PAUSE");
}
```

二、式(1)(無優化)

```
#include<iostream>
#include<math.h>
using namespace std;
main()
{
    int k,j;
```

```
double i,l,m;
cout.precision(5);
while(cin>>k)
{
    int big=0;
    double beginx=0,beginy=k,endx=0,endy=-k,na,ma;
    j=0;
    while(beginy>0)
    {
          endy=-beginy;
          while(endy<0)
          {
              j=0;
           double line[500][2]={0};
             for(i=beginy;i>=endy;i-=(beginy-endy)/k)
             {
                   line[j][0]=i;j+=1;
              }
             int q=k*k;
             int y=int(line[0][0]);
             int x=0;
             while(y<=k&&x<k)
             {
                 x=int(y-line[0][0]+0.9999);
                 q-=x;y++;
             }
             x=int(-line[j-1][0]);
             y=0;
             while(x<k&&y<k)
             {
                 q-=k-x;
                 y++;
                 x=int(y-line[j-1][0]);
             }
             //砍外圍格子
             for(int id=0;id<=k;id++)
             {
                 if(-line[id][0]>=0)
                 {
                     int u=-line[id][0],v=0;
```

```
                    int re_x=u,re_y=0,count=0;
                    while(u<=k&&v<=k)
                    {
                        if(u-v+line[id][0]>0)v++;
                        else if(u-v+line[id][0]==0)
                       {
                            count++;v++;u++;if(v>k||u>k)count--;
                       }
                        else u++;
                    }
                    q+=u-re_x+v-re_y-count-2;
                }
                else
                {
                    int v=line[id][0],u=0;
                    int re_x=u,re_y=v,count=0;
                    while(u<=k&&v<=k)
                    {
                         if(u-v+line[id][0]>0)v++;
                        else if(u-v+line[id][0]==0)
                       {
                            count++;v++;u++;if(v>k||u>k)count--;
                       }
                        else u++;
                    }
                    q+=u-re_x+v-re_y-count-2;

                }
            }
            //增內部格子
            if(big<q){big=q;na=beginy;ma=endy;}
            endy+=0.021;
        }
        beginy-=0.021;
    }
    cout<<big<<" "<<na<<" "<<ma<<endl;
  }
}
```

三、對 **k mod** 2≡1 優化之程式

其時間維度為 O(200k)

```
#include<iostream>
using namespace std;
main()
{
    int k;
    while(cin>>k)
    {
        if(k%2)
        {
            cout<<"please enter an integer that cannnot be divided by 2\n";
            continue;
        }
        double i=(k-1)/2,f,e;
        int j;
        int big=0;
        while(i<(k+1)/2+1)
        {
            int sum=k;
            for(j=1;j<=int(i)+1;j++)sum+=2*(k-j);
            sum+=(2*k-1)*(k-1);
            for(j=1;j<=(k-1)/2;j++)sum-=4*int(i*(1-2.0*j/k));
            if(sum>big){big=sum;f=i;}
            if(sum>=big){e=i;}
            i+=0.005;
        }
        cout<<big<<" from "<<f<<" to "<<e<<endl;
    }
}
```

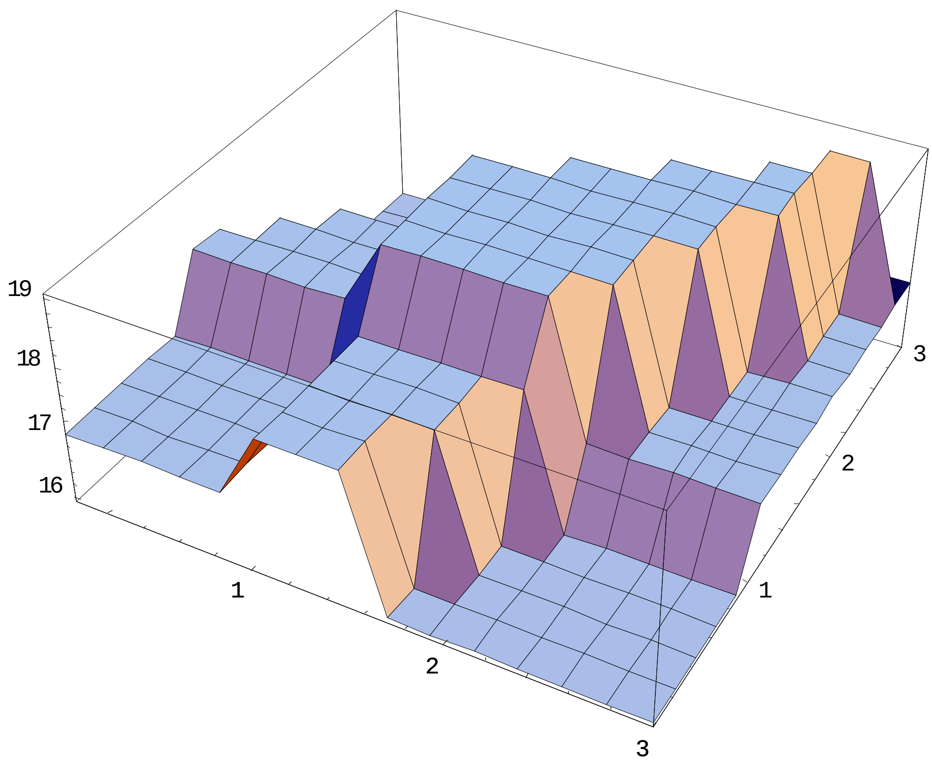
四、以猜想優化之程式

```
#include<iostream>
#include<math.h>
using namespace std;
main()
{
    int k;
    while(cin>>k)
    {
        if(k>2)cout<<(long long)(5/2.0+11/2.0*k+3/4.0*
        (pow((-1),((k-3)%2)-1))/2+(k+1)/4.0*(k-3)*9)<<endl;
        else if(k==1) cout<<1<< endl;
        else cout<<7<< endl;
    }
}
```

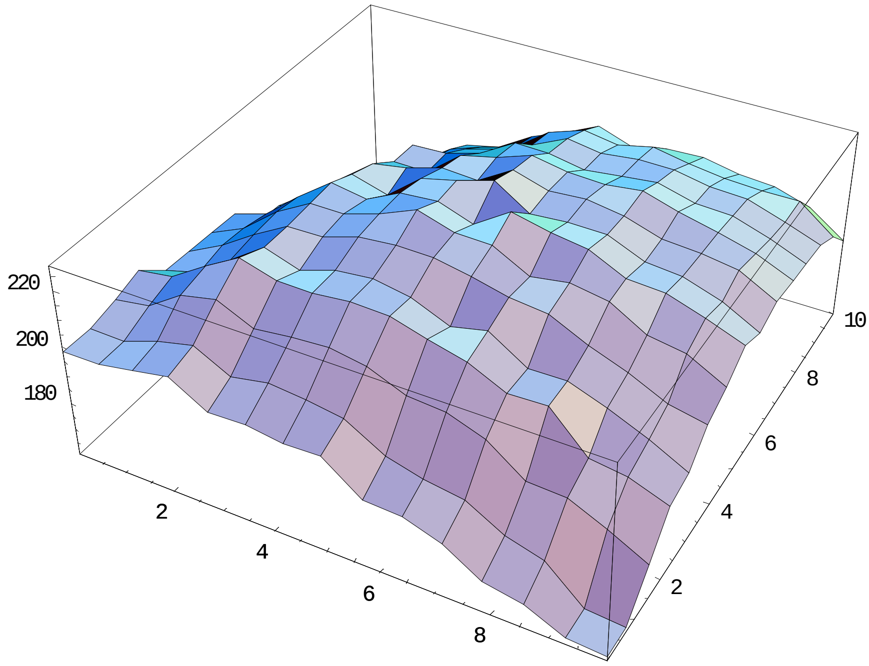
五、式(1)之圖形

X 軸,y 軸分別爲 a,b

k=3

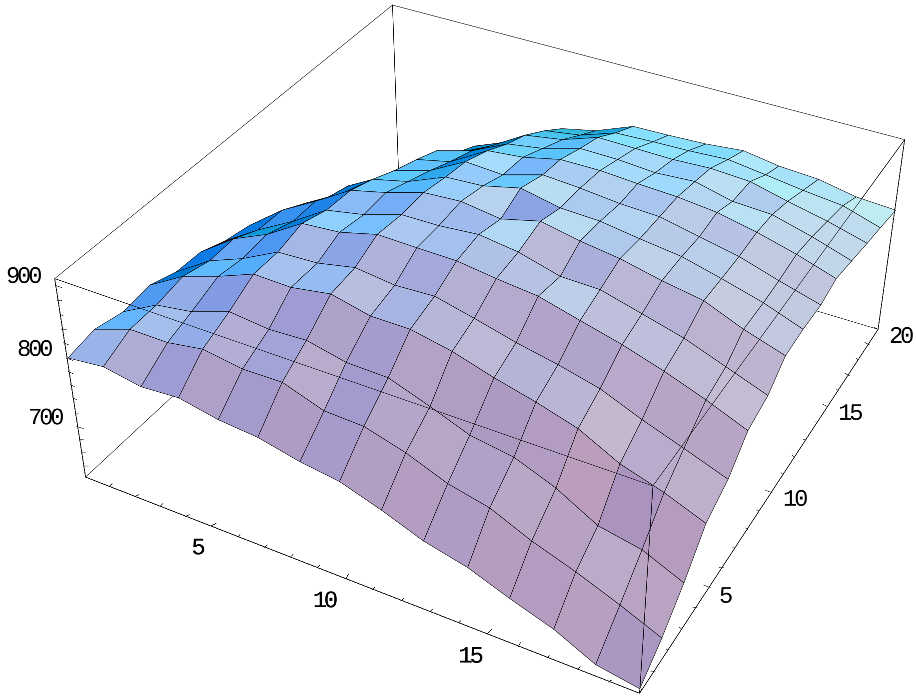


k=10



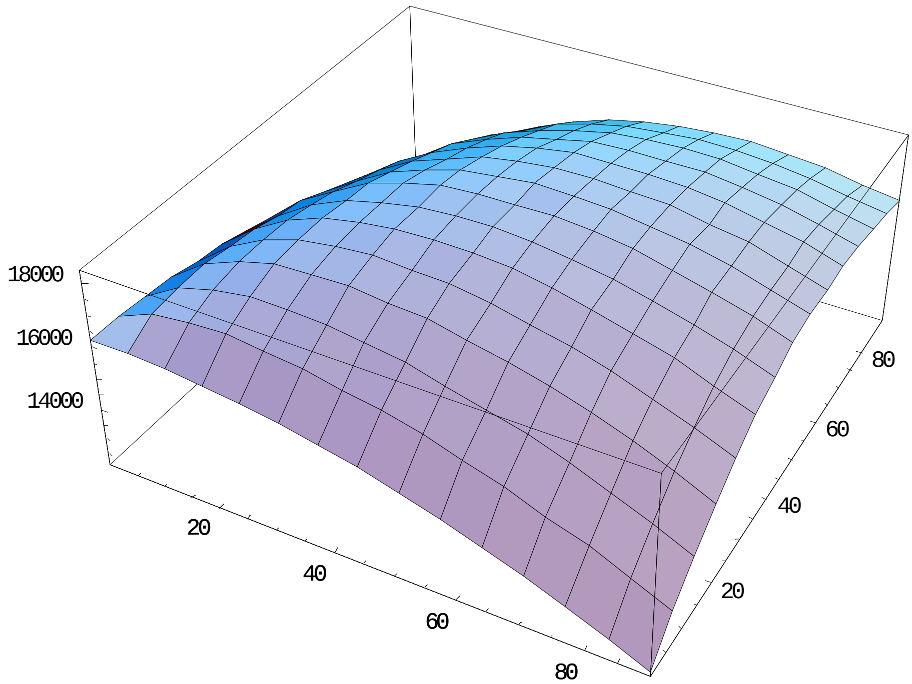
220
200
180
2
4
6
8
2
4
6
8
10

k=20



900
800
700
5
10
15
5
10
15
20

k=90



18000
16000
14000
20
40
60
80
20
40
60
80

評語

如果題目不只限於「極值之計算」，也許就會有更大的發揮空間，也應當較能展示出豐富的研究成果，無論如何，兩位的研究精神與研究態度很值得鼓勵。