

行政院原子能委員會
委託研究計畫研究報告

核反應器切割拆解最佳化設計
Optimal Dismantle Design of the Nuclear Reactor

計畫編號：1062001INER018

受委託機關(構)：國立台灣大學機械系

計畫主持人：陳湘鳳 教授

聯絡電話：02-33662692

E-mail address：ssmith@ntu.edu.tw

協同主持人：n/a

研究期程：中華民國 106 年 1 月至 106 年 12 月

研究經費：新臺幣 680,000 萬元

核研所聯絡人員：洪毓翔

報告日期：106 年 11 月 27 日

目 錄

目 錄	I
中文摘要	1
ABSTRACT.....	2
壹、計畫緣起與目的	3
貳、研究方法與過程	7
一、 從 SOLIDWORKS 讀取資料至 VISUAL STUDIO 進行最佳化設計	8
(一) 圓柱狀.....	8
1. VISUAL STUDIO 讀取 STL 檔案.....	9
(1) STL 檔案	9
2. 沿著 Y 軸計算 XZ 截面積	10
3. 進行 Y 軸座標切割.....	11
(1) 基因演算法.....	11
A. 產生初始族群 (INITIAL POPULATION)	12
B. 計算適應值 (COMPUTE FITNESS VALUE).....	13
C. 交配 (CROSSOVER).....	13
D. 突變 (MUTATION)	13
E. 選擇 (SELECTION).....	14
F. 滿足終止條件 (TERMINATION CONDITION).....	14
(2) 產生初代族群.....	14
(3) 計算適應值.....	15
(4) 染色體交配.....	16
(5) 染色體突變.....	17

(6) 染色體選擇.....	17
4. 計算角度截面積	17
5. 進行角度切割	18
(1) 產生初代族群.....	18
(2) 計算適應值.....	19
(二) 圓餅狀.....	20
1. 沿著 X 軸計算 YZ 截面積.....	21
2. 進行 X 軸座標切割.....	22
(1) 產生初代族群.....	22
(2) 計算適應值.....	23
3. 沿著 Z 軸計算 XY 截面積	24
4. 進行 Z 軸座標切割.....	25
(1) 產生初代族群.....	25
(2) 計算適應值.....	25
二、 從 VISUAL STUDIO 呼叫 SOLIDWORKS 進行自動化切割	26
(一) SOLIDWORKS 二次開發實現.....	26
1. 開發平台和開發工具選擇	26
2. VISUAL STUDIO 之必要性	27
3. SOLIDWORKS API 介面	27
4. SOLIDWORKS API 物件結構.....	28
5. SOLIDWORKS API.....	29
(1) 應用程式物件.....	29
(2) 事件物件.....	30
(3) 注解對象.....	30

(4) 模型物件	30
(5) 特徵物件	30
(6) 草圖對象	30
6. SOLIDWORKS API 對象簡介	30
(1) MODELDOC 對象	31
(2) PARTDOC 對象	31
(3) ASSEMBLYDOC 對象	31
(4) DRAWINGDOC 對象	31
(5) SELECTIONMGR 對象	32
7. 方法初步實踐及可行性	32
8. SOLIDWORKS MACRO	33
(二) 切割方式	34
1. 手動測試 BOOLEAN OPERATION	34
2. 自動實現 BOOLEAN OPERATION	37
(三) API 函數	38
1. OPENDOC2	38
2. GETBODIES2	38
3. GETMASSPROPERTIES	38
4. CREATEBODYFROMBOX	38
5. APPLYTRANSFORM	38
6. SAVEAS3	38
7. OPERATIONS2	39
8. POSTSPLITBODY	39
參、主要發現與結論	40

一、人機介面	40
(一)使用者輸入模型文件.....	40
(二)輸入切割相關參數.....	40
(三)結果分析.....	41
二、範例	41
三、限制	45
四、結論	47
肆、參考文獻	49

中文摘要

能源是現代文明發展的基石。過去，人類所需要的能源主要來自化石燃料，水電，火力，天然氣和燃煤。經過幾年建設，核能發電已經成為國家電力供應中不可或缺的一部分。

核能發電廠一般使用許可年限是 40 年左右，核一廠將於民國 107 年 12 月 5 日起停止使用，屆時原本的核反應爐就必須拆除並分解移至其它位置存放。因其大小與輻射量的限制，必須將反應爐切成數塊元件裝入制式許可的廢料容器中，而刀具的損耗是需要考量的成本，這次計畫的目的就是去尋找最佳的切割方式，來達到最少的切割面積與最少的切割刀具損耗。

本研究結合了 APRILab 的“自動化設計”和“最佳化設計”兩個研究方向，對切割的進行最佳化設計，並在 SolidWorks 中實現自動切割。

Abstract

Energy is the cornerstone of the development of modern civilization. In the past, the energy that human used was mostly from fossil fuels, hydropower, firepower, natural gas and coal combustion. With the development of nuclear energy in recent years, nuclear power generation becomes an indispensable part of the electric power supply after several years of construction.

Nuclear power plant is generally used about 40 years. ChinShan nuclear power plant will be decommissioned on use in December 5, 2018. Then the nuclear reactor must be dismantled and stored in locations. Due to its size and radiation limit, the reactor must be cut into several pieces of components into a waste container of a specified size, and the loss of the tool is the cost to be considered. The purpose of this topic is to find the best way to cut to achieve the shortest cutting path with the least cutting tool loss.

This study combines APRILab's "automated design" and "optimization design" two research directions to optimize the design of cutting, and the realization of the results of automated cutting in SolidWorks.

壹、計畫緣起與目的

各種工業設施其中也包括核設施，都會有一定的生命週期，工業設施在歷經設計、建造、試俾、運轉等各階段後，會進入除役階段進行拆除。但是核設施與一般工業設施在除役時最大的不同，是核設施的部分設備或零組件會受到中子照射而活化成為放射性廢棄物，或是因為放射性物質的附著污染而成為放射性廢棄物。基於對民眾與環境的保護，這些放射性廢棄物的處理將有別於傳統無放射性的廢棄物。

放射性廢棄物的處理程序，是先將廢棄物裝入合格的容器內，再將容器運往廢棄物貯存設施進行儲存。因此在進行放射性廢棄物的包裝之前，需要予以適當的規劃，尤其是那些大型的零組件或設備，因為其外型尺寸往往會超過容器所能容納的範圍，有時為了配合容器的尺寸，需要將一件大型廢棄物切割成多件小體積的廢棄物。因此大量的切割與包裝作業，在核設施除役過程中是不可避免的。

在進行放射性廢棄物的切割與包裝規劃之前，必須考慮下列因素，第一項因素是安全，例如要減少放射性廢棄物對於人員及環境的衝擊，某些高輻射強度的零組件需要以水作為輻射屏蔽，而採取水下切割；或是要考慮避免輻射污染的擴散，切割過程中要收集切屑及細碎顆粒，而採用熱切割(如電漿、氧

乙炔)會比機械式切割(如帶鋸機、圓盤鋸)更容易造成污染的擴散。第二項因素是成本，包括工作時間、工作人力、機具成本、耗材成本，以及切割所產生的二次廢棄物的數量(如碎屑、受污染的耗材、過濾污染的濾材)。第三項因素是容器，為了配合廢棄物的輻射強度、現場的運送限制、廢棄物貯存設施的接收標準等各項限制，可能有各種形式的合格容器可供選用，而這些容器的尺寸將影響廢棄物的切割規劃。

上述的考慮因素會影響到切割策略及工法選用，也會影響到切割規劃及包裝方式。例如切割採用機械式切割將無法進行曲線切割，切割路徑必須是直線的；又例如水下切割產生之切割件可能無法直接裝入岸上的容器，必須先在水下裝入一只提籃，當提籃裝滿在吊離水面時，必須在厚重屏蔽的包覆下，再將提籃放入容器內，此時切割件的尺寸就必須小於提籃的內部尺寸，而非容器的內部尺寸。

先前零組件的切割規劃是採用人工手動的方式，利用 CAD 軟體對於零組件 3D 模型進行除料的布林運算，將零組件切成多個切割件。因為要考慮容器的重量限制，還要讀取各個切割件的重量，作為後續包裝的參考。有些大型組件可能要細切成上百件的切割件，使得工作非常繁瑣複雜。有時在完成

切割規劃之後，又因為容器選用的改變，而必須重複此一枯燥的過程。

本研究旨在發展一套軟體，用於核設施除役時大型零組件的切割路徑規劃。又依據近期國外的除役經驗，許多高輻射零組件的水下切割，偏向使用機械式切割，例如圓盤鋸與帶鋸機，因此零組件的切割規劃路徑皆為直線。

軟體運算時需要先設定容器尺寸與刀具切縫寬度，而切縫寬度的設定可以得知切割所產生的切屑數量。再利用基因演算法進行最佳化演算，以最少的切割斷面積為目標。因為在最少切割斷面積的條件下，將使得切割件尺寸達到最大化，也使得切割時間及二次廢棄物(切屑)數量最小化，並且最大化刀具的使用壽命，進而降低在實際切割時的成本。

本研究結合了 APRILab 的「自動化設計」和「最佳化設計」兩個研究方向，對切割路徑進行最佳化的設計。軟體執行過程先讀取零組件的 3D 模型進行運算，本報告所選擇的零組件為沸水式反應器(BWR)的頂部導板(Top Guide)及反應器壓力槽，以及台灣研究用反應器(TRR)的上熱屏蔽。當最佳化演算完成後，可自動產生切割件的 3D 模型並讀取各個切割件的重量，這些切割件的模型及重量數據可用於後續的包裝規劃。

以程式運算取代人工手動進行切割路徑規劃，能夠大幅減少切割與包裝的規劃時間，有效地提升工作效率與準確性。

貳、研究方法與過程

自動化設計的部分，以 Visual Studio 作為接口，將四個模型轉換為 STL 檔案，並讀取 STL 檔案至 Visual Studio 進行截面積的判讀，接著藉由最佳化設計，得到欲切割資訊。在 Visual Studio 中使用 SolidWorks API 函數對 SolidWorks 進行二次開發，將得到的切割資訊回送至 SolidWorks 中，於 SolidWorks 內自動對模型進行切割。

最佳化設計使用的是基因演算法，基因演算法是人類依照生物學中「適者生存，不適者淘汰」的觀念所發展出來的一種啟發式演算法利用「選擇、複製」、「交配」、「突變」等步驟去尋找最適合環境的基因。本研究尋求在 SolidWorks 中進行自動化切割的實現結果。程式環境為 Windows 7, SolidWorks 2016, 及 Visual Studio 2015，

本研究主要分為兩個部分：從 SolidWorks 讀取資料至 Visual Studio 進行最佳化設計，以及從 Visual Studio 呼叫 SolidWorks 進行自動化切割。方法流程圖如圖 1 所示。

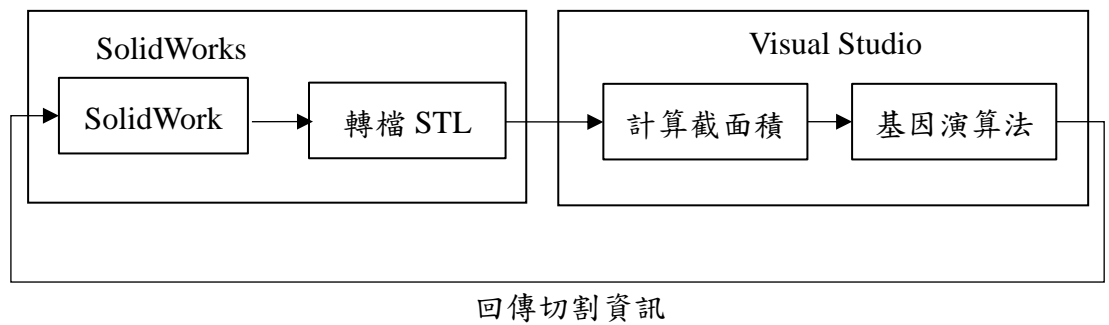


圖 1、方法流程圖

一、從 SolidWorks 讀取資料至 Visual Studio 進行最佳化設計

由核研所提供的模型可分為圓柱狀和圓餅狀，分別對兩種形狀模型進行切割最佳化設計。

(一)圓柱狀

圓柱狀最佳化設計的流程如圖 2 所示。

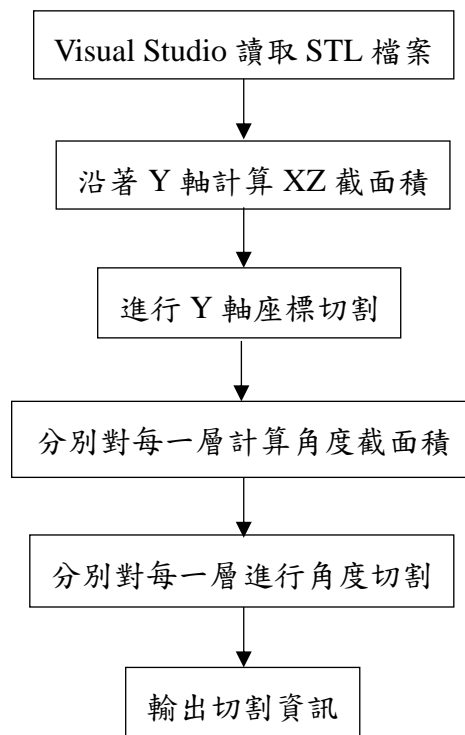


圖 2、圓柱狀最佳化設計流程圖

1. Visual Studio 讀取 STL 檔案

我們使用以模型中心為原點，將模型轉換成 STL 二進制模型檔案，接著再以 Visual Studio C#讀取來判斷模型的資訊。使用的是 C#中 System.IO 類別的 BinaryReader 來讀取二進制 STL 檔案中的檔案名稱、三角形數量及每一個三角形的頂點、法向量。

(1) STL 檔案

CAD 系統與 RP 系統的資料交換介面格式 STL，是美國 3D System 公司於 1987 年提出的，目前已成為 RP 工業界的標準。一個三維實體模型的 STL 檔是用大量的空間小三角形來表示實體模型的表面。對每一個空間小三角形，用三角形的三個頂點及法向量來描述。法向量由實體的內部指向外部，三個頂點的次序與法向量滿足右手規則。此外，STL 檔中兩個相鄰的三角形只能有一個公共邊。STL 檔有兩種格式：一種是 ASCII 格式，另一種是二進位格式。二進位格式如下：

UINT8[80] – Header

UINT32 – Number of triangles

For each triangle

REAL32[3] – Normal vector

REAL32[3] – Vertex 1

REAL32[3] – Vertex 2

REAL32[3] – Vertex 3

UINR16 – Attribute byte count

End

2. 沿著 Y 軸計算 XZ 截面積

首先找到模型最高及最低點，得到模型高度。從最高點開始到最低點以 10 mm 為間距，計算每一處所經過的三角形距離原點最大與最小值，如圖 3 所示。最大最小值相減便是此處截面積數值，並創造一陣列記錄每一處的截面積值。

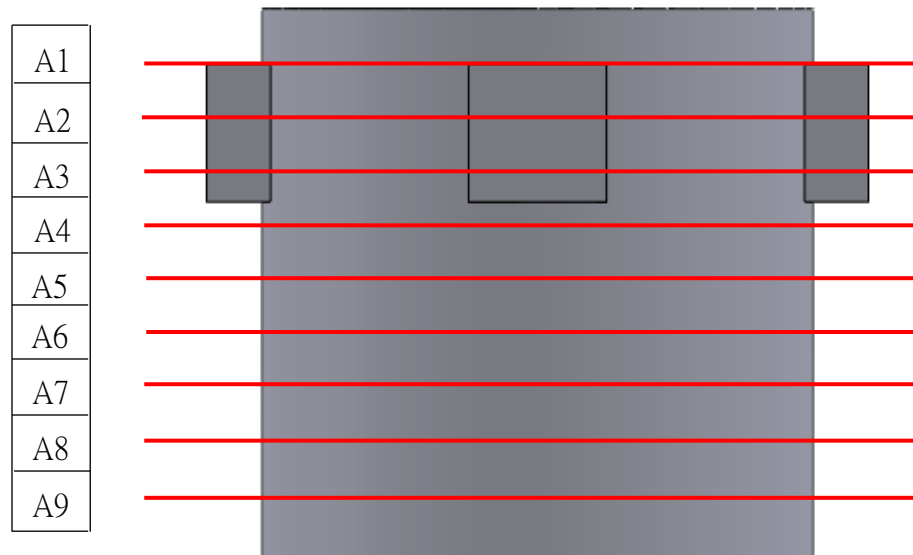


圖 3、沿著 Y 軸計算 XZ 截面積

3. 進行 Y 軸座標切割

(1) 基因演算法

本研究使用的切割演算法為基因演算法 [1]。基因演算法(Genetic Algorithms；GA)或稱為遺傳演算法，由美國密西根大學 John Holland 教授所提出來的一種演化式 (Evolutionary Computing)演算法。他是根據生物學家達爾文 (Charles Darwin) 所提出的進化論「物競天擇，適者生存」的概念，進一步發展而來。其主要利用電腦模擬生物的競爭—「複製、交配、突變」保留適應力較佳的染色體，來繁衍最優越的下一代。圖 4 為基因演算法流程圖。

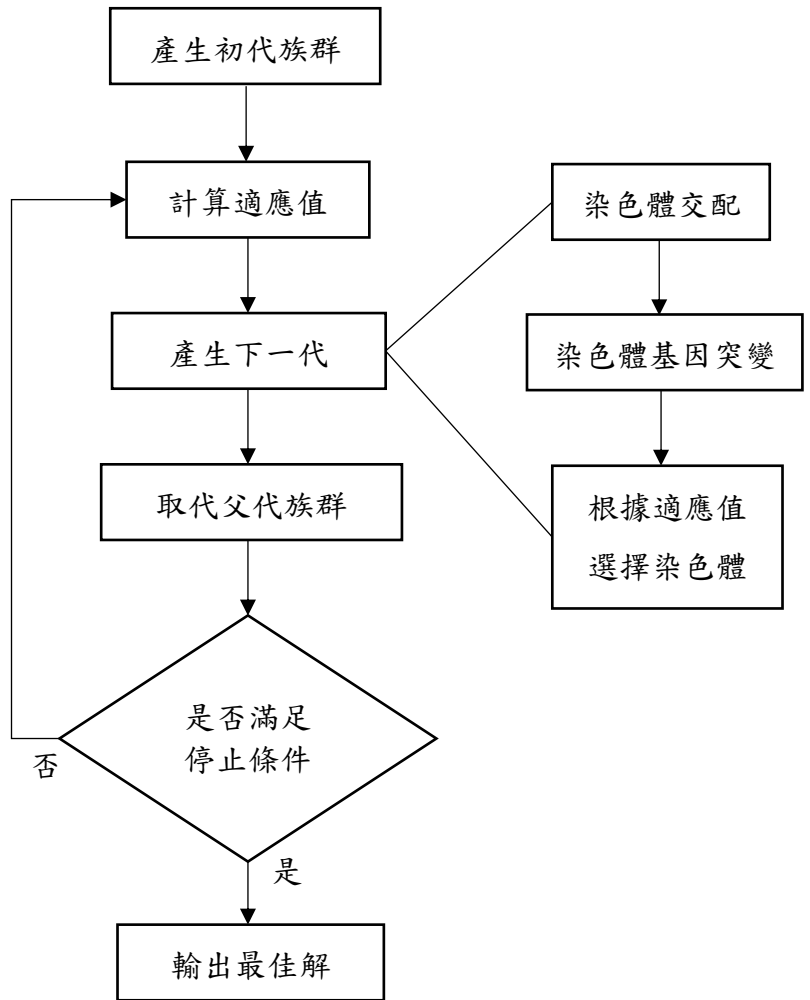


圖 4、基因演算法流程圖

A. 產生初始族群 (Initial population)

基因演算法中，初始族群的族群大小 (population size) 對於求解效益有很大的影響。族群太大有較大機會達成最佳解，但耗費計算較高，需要較多演算時間。族群太小容易陷入局部最佳解，有提早收斂之可能。

B. 計算適應值 (Compute fitness value)

適應值主要是用來判定族群染色體中適應環境的能力，所以一個好的適應性函數，要能夠將染色體的好壞判別出來。根據不同的問題，發展出不同的適應性函數，在基因演算法中，適應性函數的目的是判斷某一組染色體的適應程度，最終的目的就是要找到適應值最高的染色體。

C. 交配 (Crossover)

交配為基因演化的核心機制之一，主要目的是讓挑選出來兩個染色體交換彼此的基因，以產生更優良的基因組合。一個染色體的進化，也許需藉助要其他染色體上的基因功能才能補全，因此必須藉由交配來達成以上目的。

D. 突變 (Mutation)

突變是基因演算法中相當特殊的機制，也是提供一個跳出目前搜尋空間避免落入局部最佳解的方式。突變機制一般來說目的有二，其一為開發新的搜尋區域，使求解方向產生無限可能；另一則為重新導入群體在演化過程中遺

失的重要資訊。

E. 選擇 (Selection)

選擇族群中的染色體，使適應值較佳的染色體被保留下來，較差的染色體被淘汰。所以在下一子代的族群中，適應值較低的染色體將逐漸減少，而取而代之的則是適應值較高的染色體，因此族群中便產生最優良的子代染色體。

F. 滿足終止條件 (Termination condition)

基因演算法之終止條件可以分為以下三種：演化代數、達到目標、或系統收斂。設定基因演算多少代則停止，此優點為能控制計算時間，但無法確定是否收斂能夠找到最佳解。設定目標值，若已經達到則停止。此方式能確保求解的品質，但是必須要先知到目標的範圍，否則演化永遠不會結束。若連續很多個群體，在沒有突變的情況下群體的個體都相同，在歷經好幾代的適應值都相同，表示已經收斂到某個程度，可能是最佳解。

(2) 產生初代族群

基因演算法的設計，設計一個足夠長度的染色體，染色體中每一個值代表切割高度，如圖 5 所示。並且為小於等於裝箱容器高度(公分)的隨機整數。當切割高度總和大於模型高度時，則調整使切割高度總和等於模型高度，並將之後隨機分配每個切割高度值歸零。

$h_1 \sim h_n$ 為小於等於裝箱容器高度(公分)的隨機整數

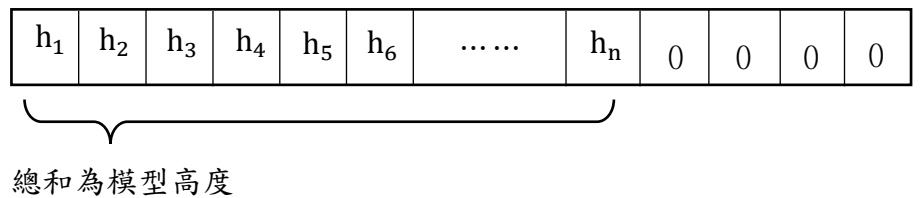


圖 5、Y 座標切割的染色體設計

(3) 計算適應值

染色體的適應值為染色體中非零整數數量加上切割高度對應的截面積總合，如圖 6 所示。

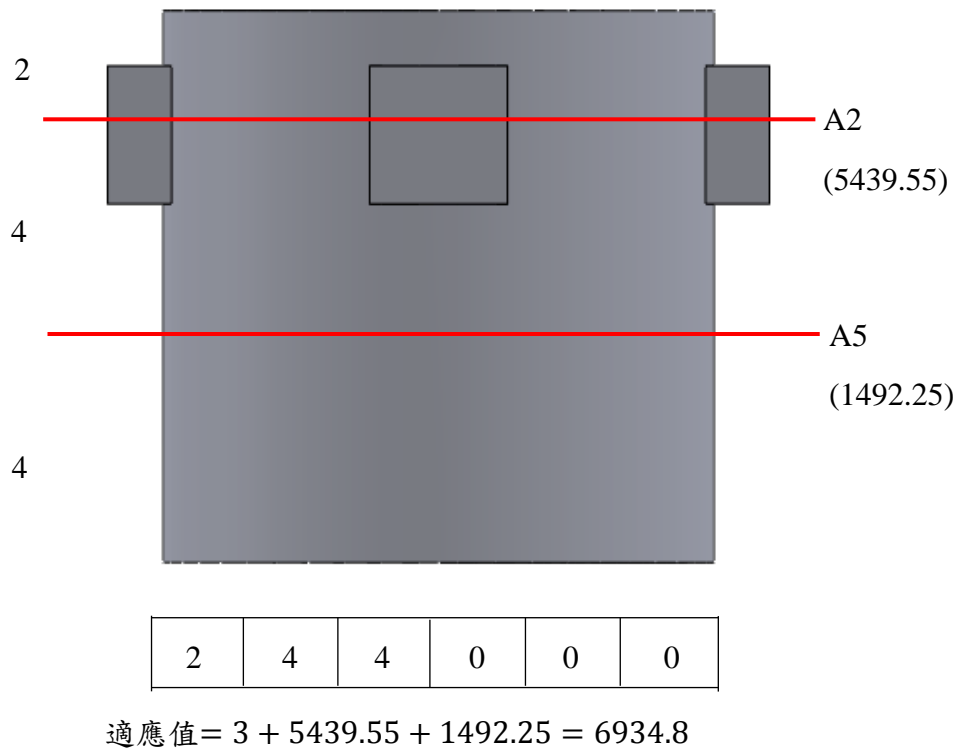


圖 6、Y 軸座標切割染色體的適應值計算

(4) 染色體交配

如圖 7 所示。

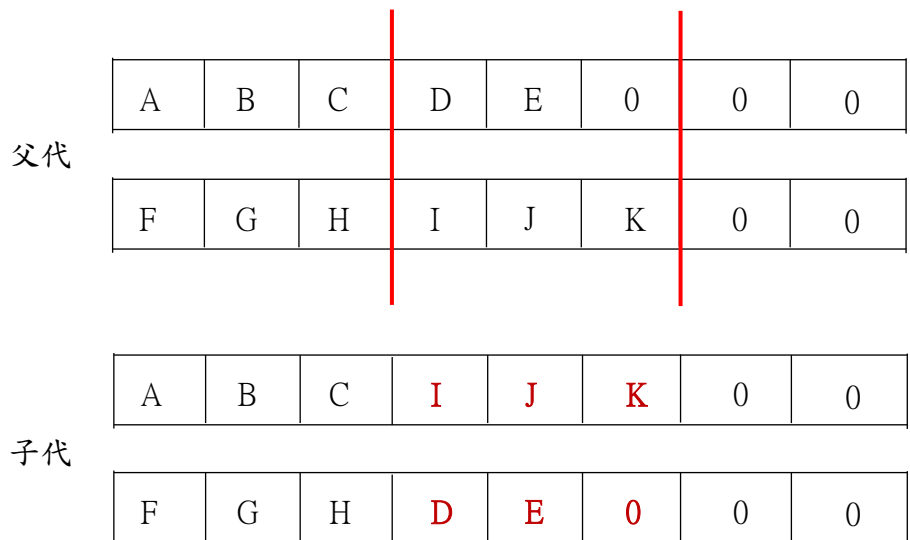


圖 7、染色體交配

(5) 染色體突變

如圖 8 所示。

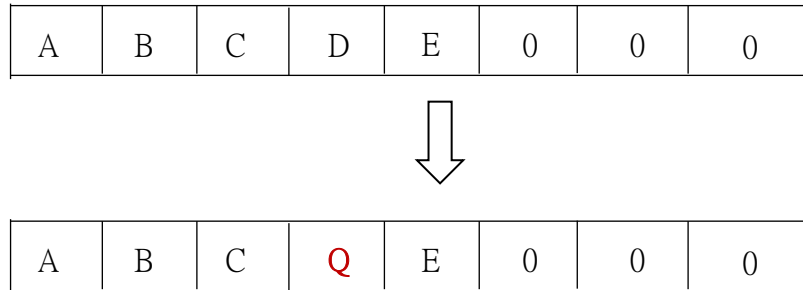


圖 8、染色體突變

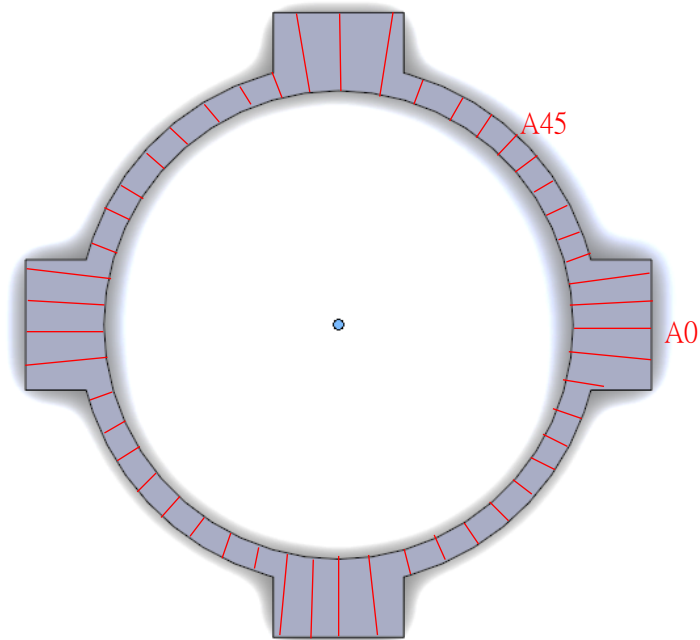
(6) 染色體選擇

選擇適應值小的染色體作為下一代。

4. 計算角度截面積

執行完 Y 座標切割後，分別對於每一層元件進行計算角度截面積。從最 0 度開始到 359 度以 1 度為間距，計算每一度所經過的三角形距離圓心最大與最小值，如圖 9 所示。最大最小值相減，再乘上元件高度，便是此角度截面積數值，並創造一陣列記錄每一度的截面積值。

A0	A1	A2	A3	A4	...	A358	A359
----	----	----	----	----	-----	------	------



$$A0 = (60 - 45) \times t = 15t$$

$$A45 = (50 - 45) \times t = 5t$$

t 為 Y 軸切割後的結果

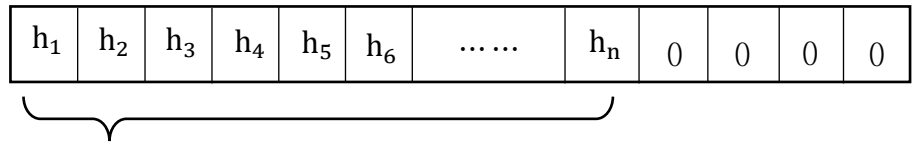
圖 9、計算角度截面積

5. 進行角度切割

(1) 產生初代族群

設計一個足夠長度的染色體，染色體的第一個值表示切割起始角，接著每個值代表切割角度，並且為小於等於最大切割角度的隨機整數，如圖 10 所示。最大切割角度由裝箱容器長度求得，如圖 11 所示。當切割角度和大於 360 度時，則調整使切割角度和等於 360 度，並將之後隨機分配每個切割角度的值歸零。

$h_1 \sim h_n$ 為小於等於裝箱容器高度(公分)的隨機整數



總和為模型高度

圖 10、角度切割的染色體設計

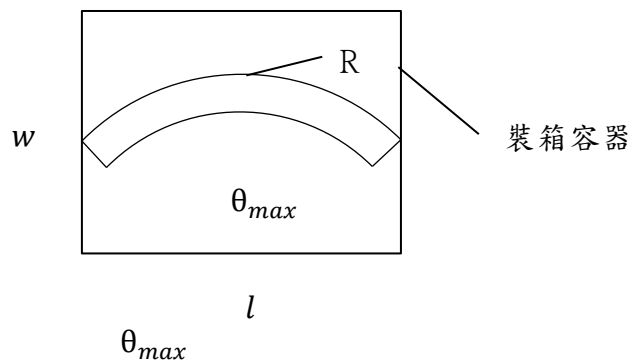


圖 11、最大切割角度

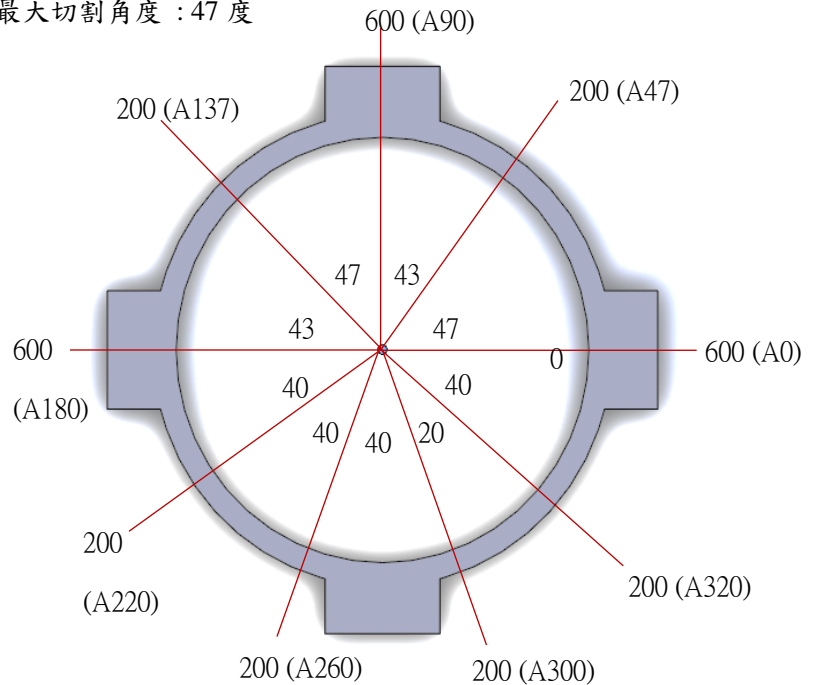
(2) 計算適應值

染色體的適應值為切割角度數量加上切割角度對應的截面積總合，如圖 12 所示。進行交配及突變後，選擇適應值小的子代當作父代進行下一次繁衍，直到適應值收斂為止，便完成一層的角度切割。

裝箱容器長度：40mm

模型半徑：50mm

最大切割角度：47 度



0	47	43	47	43	40	40	40	20	40	0
---	----	----	----	----	----	----	----	----	----	---

$$\text{適應值} = 9 + 600 + 200 + 600 + 200 + 600 + 200 + 200 + 200 + 200 = 3009$$

圖 12、角度切割染色體的適應值計算

(二) 圓餅狀

圓柱狀最佳化設計的流程如圖 13 所示。

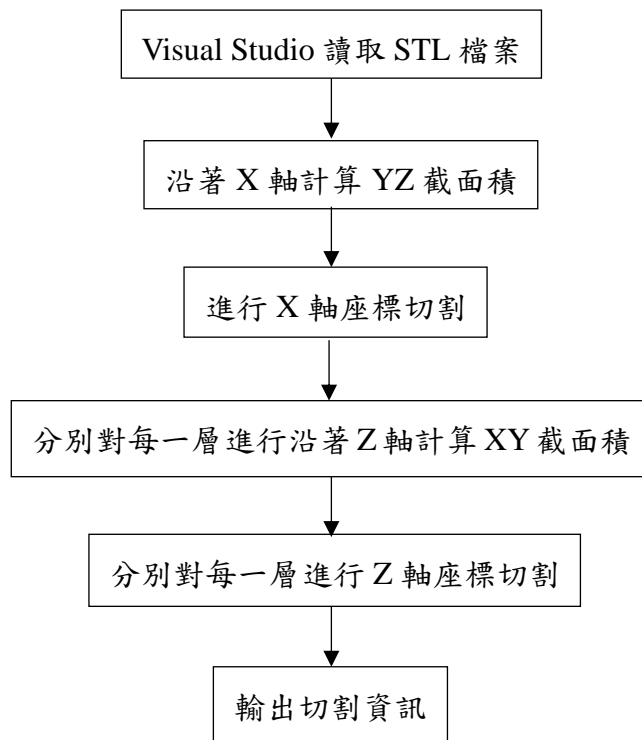


圖 13、圓餅狀最佳化設計流程圖

1. 沿著 X 軸計算 YZ 截面積

如同圓柱狀 Y 軸截面積，找到模型 X 軸最大值及最小值。從最大值開始到最小值以 10mm 為間距，計算每一處所包含的三角形 Z 軸長度，如圖 14 所示。將所有的長度相加便是此處截面積數值，並創造一陣列記錄每一處的截面積值。

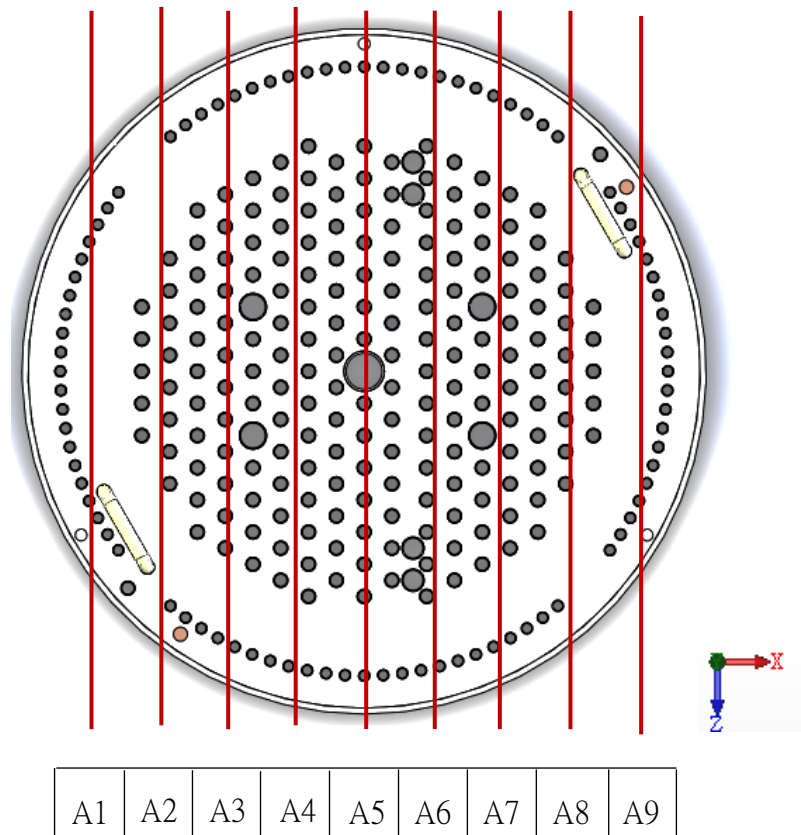


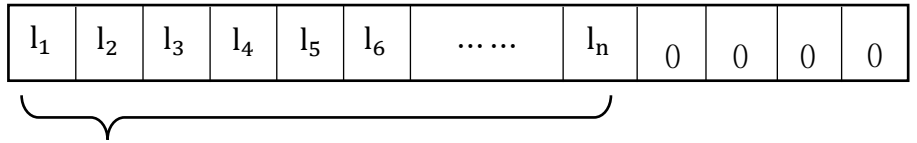
圖 14、沿著 X 軸計算 YZ 截面積

2. 進行 X 軸座標切割

(1) 產生初代族群

設計一個足夠長度的染色體，染色體中每一個值代表切割長度，並且為小於等於裝箱容器長度(公分)的隨機整數，如圖 15 所示。當切割長度和大於模型長度時，則調整使切割長度和等於模型長度，並將之後隨機分配每個切割長度的值歸零。

$l_1 \sim l_n$ 為小於等於裝箱容器長度(公分)的隨機整數

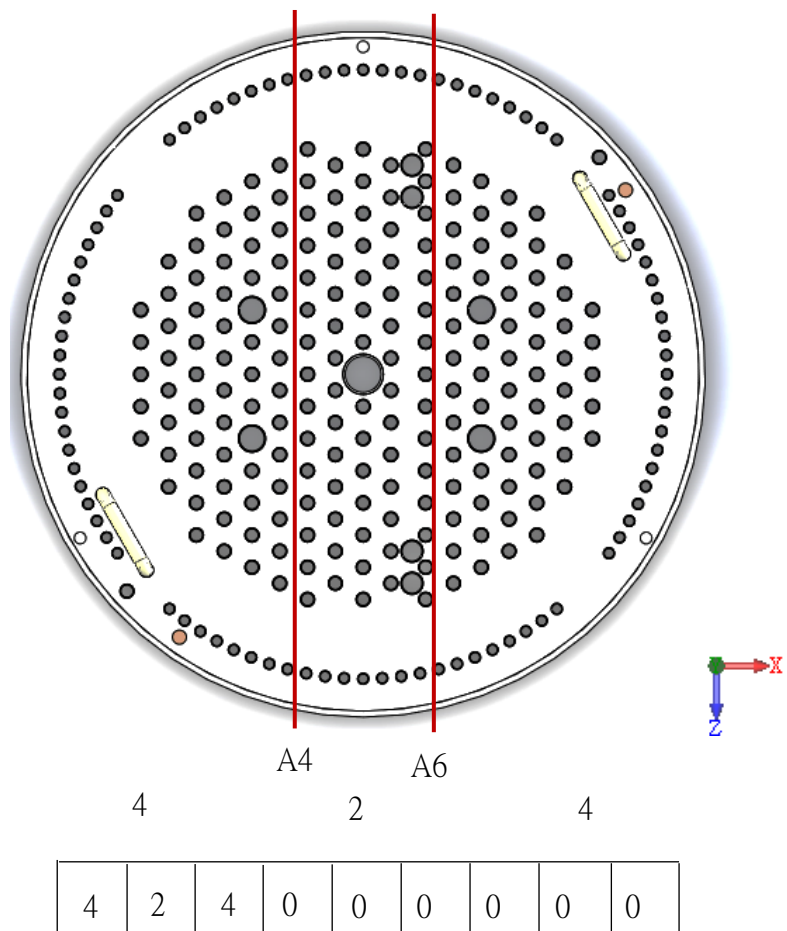


總和為模型長度

圖 15、X 軸座標切割的染色體設計

(2) 計算適應值

染色體的適應值為染色體中非零整數數量加上切割長度對應的截面積總合，如圖 16 所示。進行交配及突變後，選擇適應值小的子代當作父代進行下一次繁衍，直到適應值收斂為止，便完成 X 座標切割。



$$\text{適應值} = 3 + A4 + A6$$

圖 16、X 軸座標切割染色體的適應值計算

3. 沿著 Z 軸計算 XY 截面積

如同圓柱狀 X 軸截面積，找到模型 Z 軸最大值及最小值。從最大值開始到最小值以 10 mm 為間距，計算每一處所包含的三角形 X 軸長度，如圖 17 所示。將所有的長度相加便是此處截面積數值，並創造一陣列記錄每一處的截面積值。

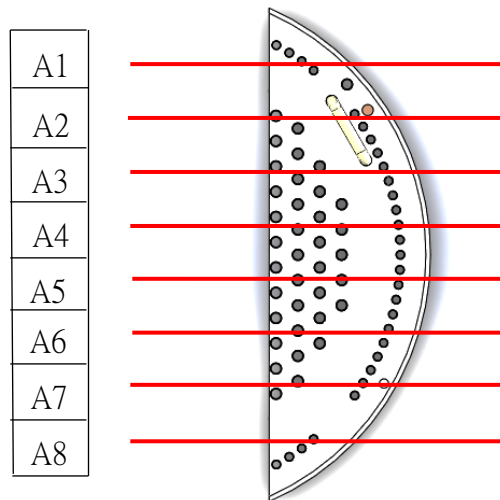


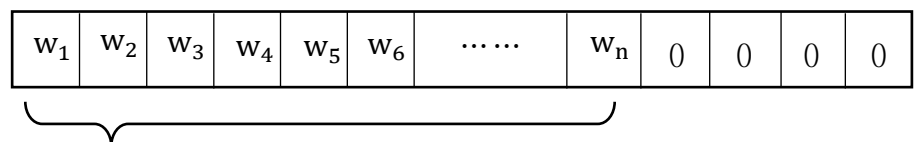
圖 17、沿著 Z 軸計算 XY 截面積

4. 進行 Z 軸座標切割

(1) 產生初代族群

設計一個足夠長度的染色體，染色體中每一個值代表切割寬度，並且為小於等於裝箱容器寬度(公分)的隨機整數，如圖 18 所示。當切割寬度和大於模型寬度時，則調整使切割角度和等於模型寬度，並將之後隨機分配每個切割寬度的值歸零。

$w_1 \sim w_n$ 為小於等於裝箱容器寬度(公分)的隨機整數



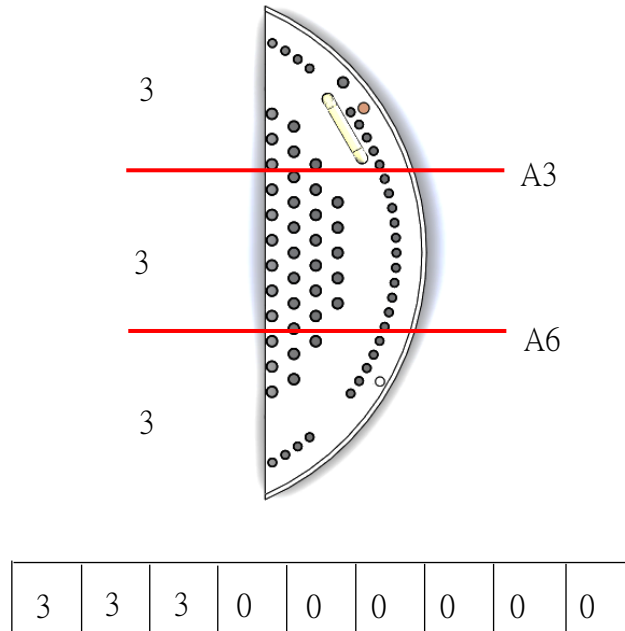
總和為模型寬度

圖 18、Z 軸座標切割的染色體設計

(2) 計算適應值

染色體的適應值為非零整數數量加上切割寬度對

應的截面積總合，如圖 19 所示。進行交配及突變後，選擇適應值小的子代當作父代進行下一次繁衍，直到適應值收斂為止，便完成一層的 Z 座標切割。



$$\text{適應值} = 3 + A3 + A6$$

圖 19、Z 軸座標切割染色體的適應值計算

二、從 Visual Studio 呼叫 SolidWorks 進行自動化切割

這個部分是使用 Visual Studio 對 SolidWorks 進行二次開發。

(一) SolidWorks 二次開發實現

1. 開發平台和開發工具選擇

目前主流的三維設計軟體有很多種類，如：CATIA、UG、Pro/E、AutoCAD、SolidWorks、I-DEAS 等。由於客戶提供的是在 SolidWorks2016 設計的實體反應器壓力槽，SolidWorks 軟體自帶有 VBA 二次開發工具，且該三維設計軟體是在 Visual C++開發的，因此若是採用

Visual Studio 二次開發，二次開發的模組可以完全嵌入到 SolidWorks 中而不會發生被排斥現象。

2. Visual Studio 之必要性

SolidWorks 具有自身 VBA 開發工具。所有執行的命令與操作都可以被 Macro 巨集所記錄，並調整。數千個 API 函數，將命令封裝，可以直接調用函數執行自動化命令之操作。SolidWorks 之 VBA 局限性，控制項有限，無法深度開發。可行的方法，除了 VBA 之外，基於 Visual Studio 的二次開發是建立在 Visual Studio 與 SolidWorks 的合法連結與正確設置。

3. SolidWorks API 介面

正確調用 SolidWorks API 是完成 SolidWorks 二次開發中參數化設計的基礎。調用 SolidWorks API 是指調用 SolidWorks 的事件、方法、屬性、以及相關功能，從而完成零部件實體建模的全過程。

SolidWorks 的 API 介面分為兩種：一種是基於 OLE Automation 的 IDispatch 技術；另一種是基於 Windows 基礎的 COM。OLE(Object Linking and Embedding，物件連接與嵌入)，簡稱 OLE 技術。基於 OLE Automation 的 IDispatch 技術作為快速開發的手段，

一般常用於 VB、Delphi 程式設計語言的介面。通過 IDispatch 介面暴露物件的屬性和方法，以便在客戶程式中使用這些屬性並調用它所支援的方法。此種技術只能開發 EXE 形式的程式，所開發的 CAD 系統不能直接加掛在 SolidWorks 系統介面下，無法實現與 SolidWorks 系統的集成。

COM (Component Object Model, 元件物件模型) 技術是 SolidWorks API 的基礎，是 Microsoft 公司提出的並被大多數公司支援的一種標準協議，它建立了一個軟體模組同另一個軟體模組的連接，當這種連接建立之後，兩個模組之間就可以通過介面來進行通訊。COM 介面更為簡潔高效，這種技術可以使用最多的 SolidWorks API 函數。

4. SolidWorks API 物件結構

SolidWorks API 通過物件導向思想組織所有的介面物件，如圖 20 所示。

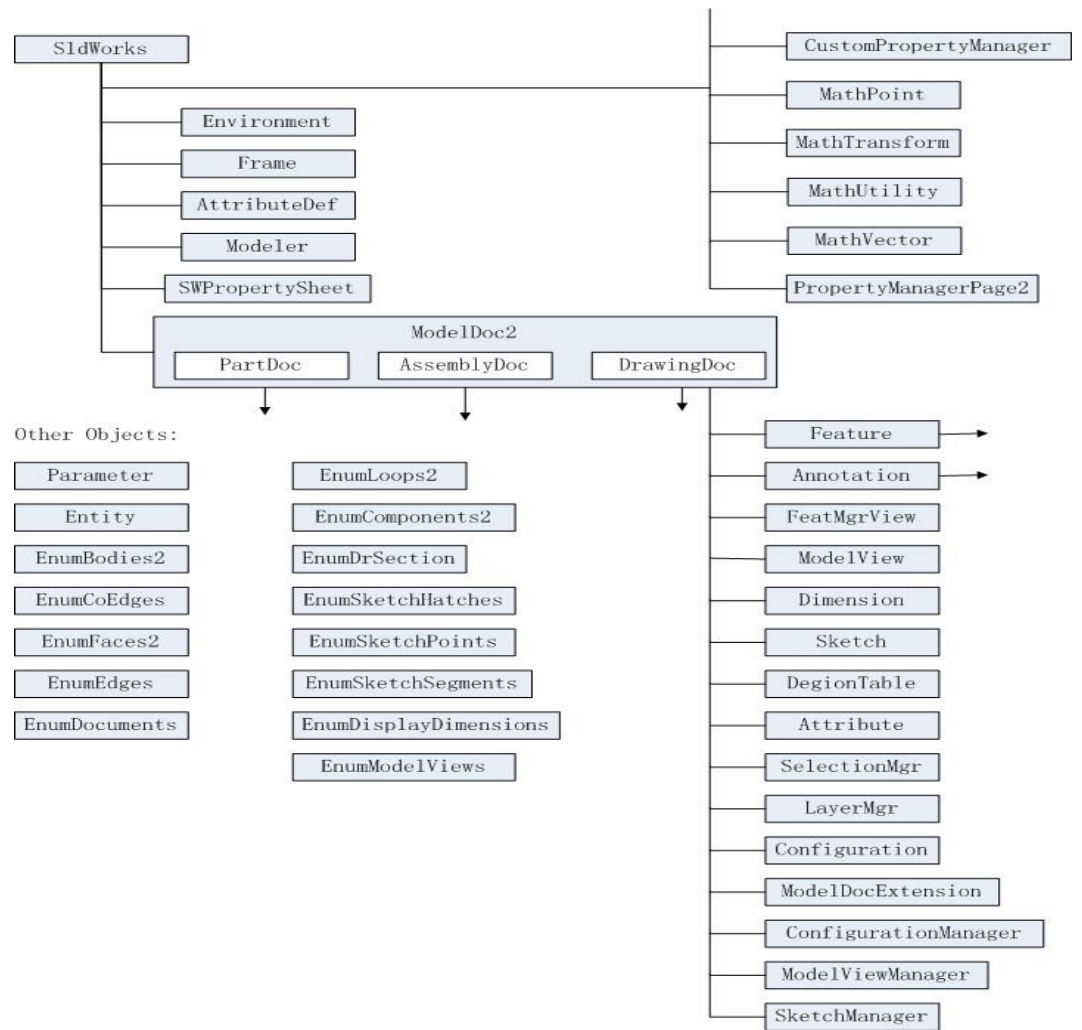


圖 20、SolidWorks API 物件結構

5. SolidWorks API

(1) 應用程式物件

應用程式物件包括 SolidWorks、ModelDoc2、PartDoc、AssemblyDoc 及 DrawingDoc 對象。設定檔對象：設定檔物件管理零件中不同模組（零件文檔模式）與裝配體中不同零件（裝配體文檔模式）的狀態。

(2) 事件物件

SolidWorks API 介面中提供了對事件的支援，當前版本中支援的事件類型有 AssemblyDoc 事件、DrawingDoc 事件、FeatMgrView 事件、ModelView 事件、PartDoc 事件、SldWorks 事件及 SWPropertySheet 事件。

(3) 注解對象

注解物件管理文檔的注解。

(4) 模型物件

描述 SolidWorks 內部資料結構。

(5) 特徵物件

描述 SolidWorks 應用程式所提供的特徵操作，這些特徵物件與 SolidWorks 軟體本身提供的特徵操作相對應。

(6) 草圖對象

包括管理所有的草圖元素，如圓弧、長方形、曲線等。

6. SolidWorks API 對象簡介

SolidWorks 物件是所有其他物件的父類，提供了直接或間接訪問其他所有 SolidWorks API 介面的方法，

在二次開發中訪問其他介面都要通過它。SolidWorks 物件是二次開發中最重要的一個物件，通過它才能建立二次開發外掛程式與 SolidWorks 應用程式之間的連接。

(1) ModelDoc 對象

ModelDoc 物件是 SolidWorks 物件的子物件，同時也是所有文檔模型物件（PartDoc 物件、AssemblyDoc 物件及 DrawingDoc 物件）的父物件，它封裝了不同文檔模型通用的屬性和方法，包括文檔開啟、關閉、列印和保存等。同時，ModelDoc 物件提供了直接訪問 PartDoc 物件、AssemblyDoc 物件和 DrawingDoc 物件的方法。

(2) PartDoc 對象

PartDoc 物件提供 Part 文檔模式下的操作。

(3) AssemblyDoc 對象

AssemblyDoc 物件管理裝配體的操作時與裝配相關的，如導入零部件、添加/取消裝配關係、隱藏/現實零部件等。

(4) DrawingDoc 對象

DrawingDoc 物件管理工程圖文檔的操作，如創

建、刪除、對其視圖等。

(5) SelectionMgr 對象

SelectionMgr 物件是選擇管理物件，用於管理使用者的選擇操作，通過它可以獲得指向當前用戶在 SolidWorks 中選擇的元素。

在實際使用中，由於 API 有數百上千個函數，每個都去熟悉瞭解不現實。因此在實際中，借助於 SolidWorks 自帶的 Macro 巨集命令進行錄製手工操作過程，打開編輯 Macro，使用逐行調試的方式，對實際操作的 command 進行瞭解，再借助於 online help 或者 SolidWorks 官方論壇進行進一步學習理解其功能及參數設置的含義。

7. 方法初步實踐及可行性

前人在針對各家公司各自所需，採用 Visual C++6.0 平台與 SolidWorks 開發的研究相當廣泛，資料也是相當多。但對於當今的作業系統平台已經發展到了 64 位元的 Windows7，原有的 32 位元平台開發的資料已經不適應新平台的使用。且各家公司針對各自公司的特殊需求，所深入研究所需的 API 函數，且 DS 公司也沒有專門針對哪家公司進行的二次開發培訓，所以使用 Macro 對 API 函數嘗試學習是一條重要的途徑。圖 21 為設置

Macro 按鈕之範例。

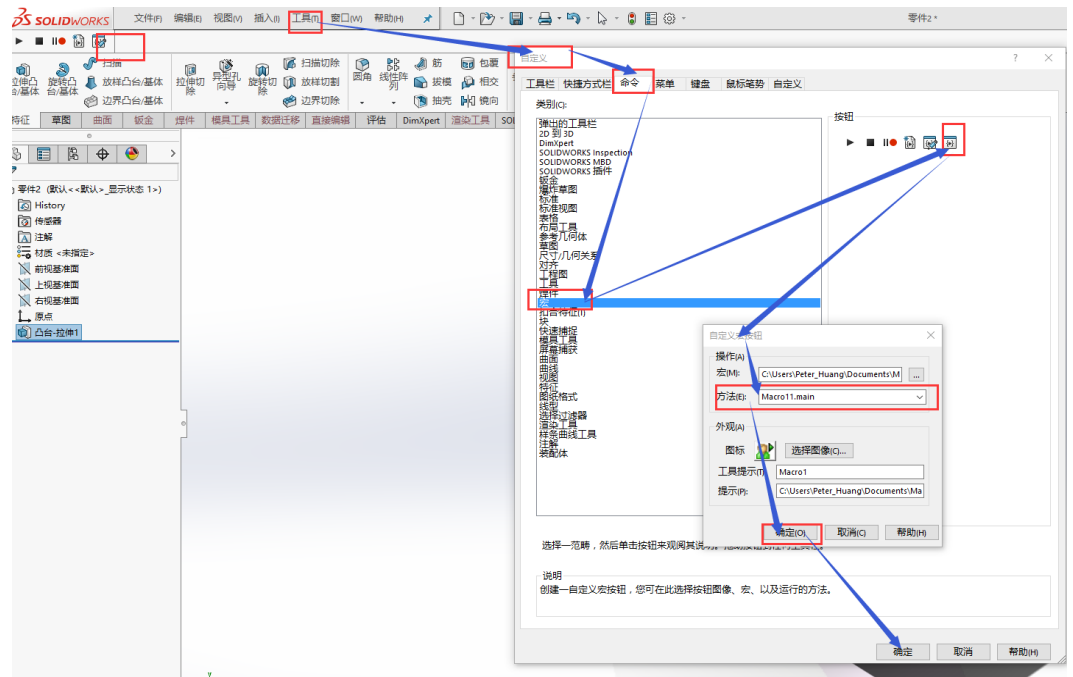


圖 21、設置 Macro 按鈕

8. SolidWorks Macro

借助於 SolidWorks 自帶的 Macro 巨集記錄工具，可以直接在 SolidWorks 自帶的 VBA 中測試所要使用的操作命令，刪除冗餘的命令列，將記錄下來的有效代碼，結合線上 SolidWorks API 說明，學習函數的正確參數設置及運行，如圖 22 所示 [2, 3]。在代碼行設置中斷點；breakpoint，使用 F5 執行調試；並 F8 進行逐行調試，觀看 SolidWorks 的運行展示。在 SolidWorks 中，也允許客戶創建使用者介面：以在 VBA 中打開“插入”—使用者表單方式進行設計。圖 23 為

SolidWorks 中 VBA 介面。

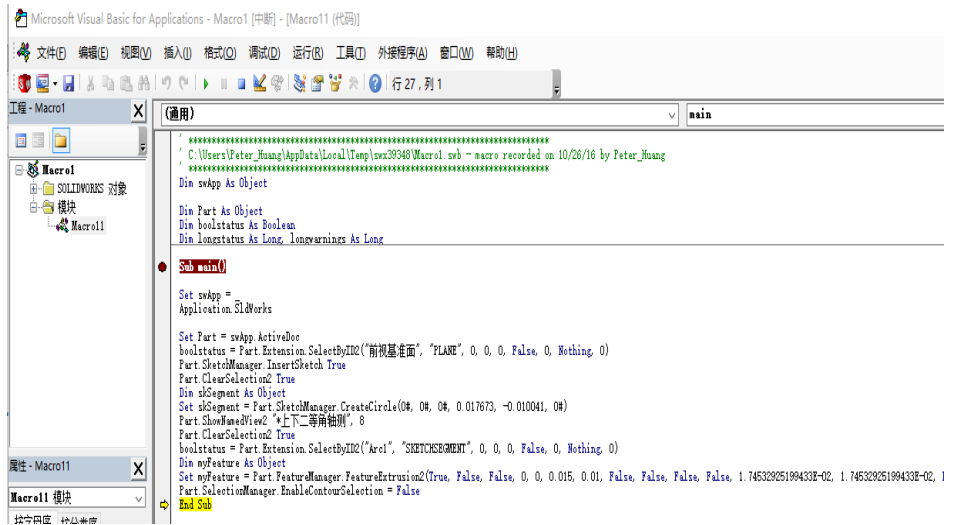


圖 22、SolidWorks 中 VBA 代碼

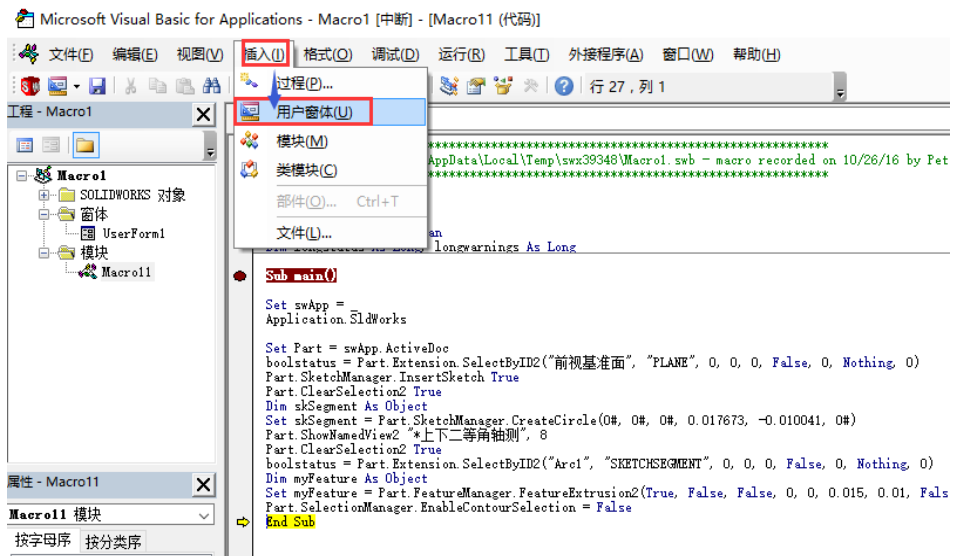


圖 23、SolidWorks 中 VBA 介面

(二) 切割方式

1. 手動測試 Boolean operation

(1) 創建圓柱體，及模擬切削用，具有一定厚度的輔助

平面實體，如圖 24 所示。

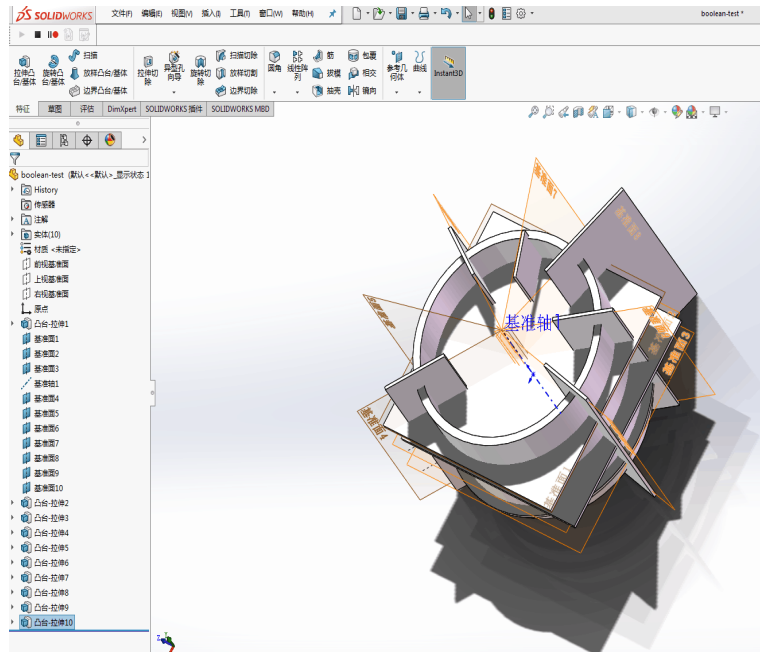


圖 24、創立圓柱及輔助平面

(2) 手動操作中，進行 Boolean operation。選擇“插入”——“特徵”——“組合”——“刪減”，選擇主要保留的實體，以及需要組合的實體，如圖 25 所示。

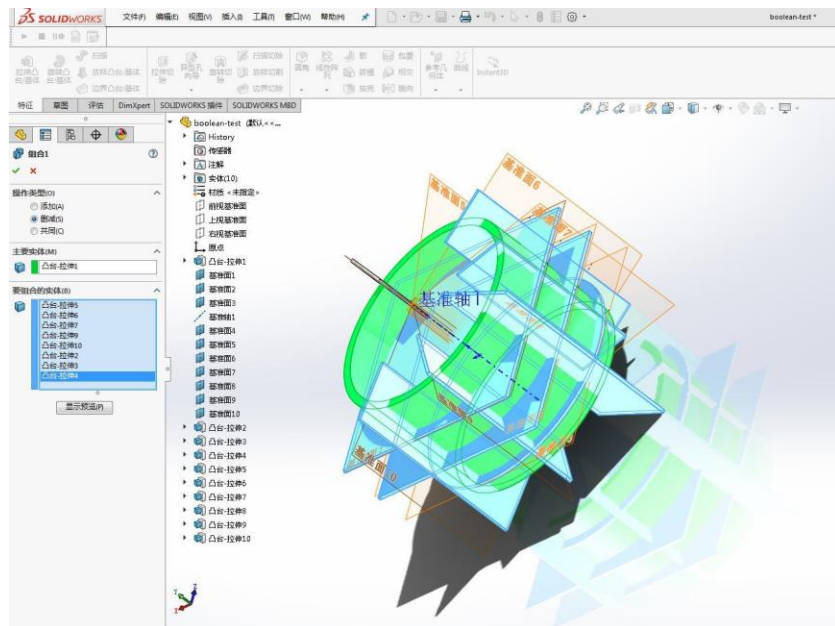


圖 25、進行 Boolean operation

(3) 選擇“要保留的實體”，如圖 26 所示。操作後的結果如圖 27 所示。

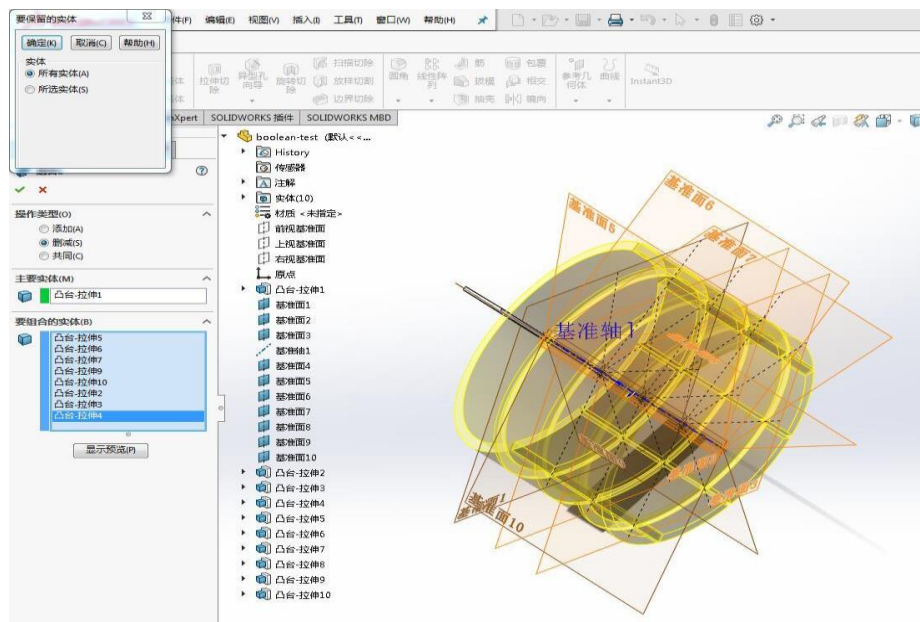


圖 26、選擇保留的實體

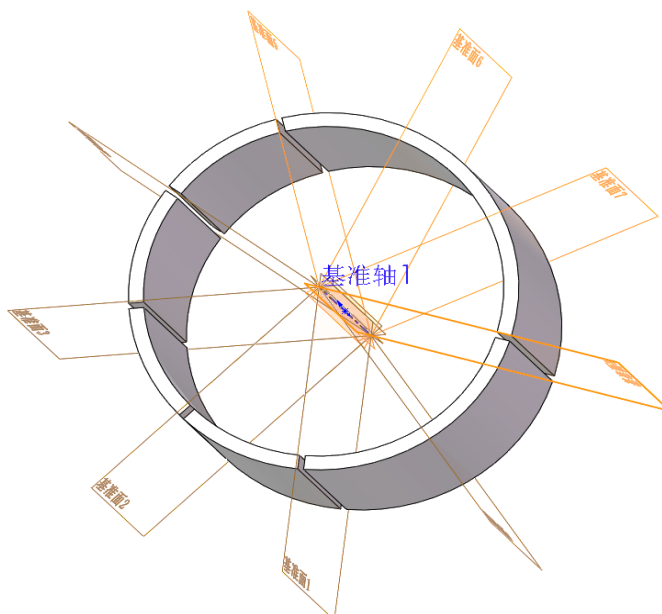


圖 27、操作結果

2. 自動實現 Boolean operation

在 Visual studio 中加入 SolidWorks API 函數，自動實現 Boolean operation 之程式碼如下。

(1) 開啟指定模型檔案

```
model = swApp.OpenDoc2(dlg.FileName, (int)swDocumentTypes_e.swDocPART, false, false, true, ref istatus);
```

(2) 讀取模型 body

```
Object[] objects = part.GetBodies2((int)swBodyType_e.swSolidBody, false);
```

(3) 創造具有一定厚度的輔助平面模擬刀具

```
private void CreatingCutBody()
{
    double length = 40;
    double width = 40;
    double[] myBox = new double[9];
    myBox[0] = 0; myBox[1] = 0; myBox[2] = 0;
    myBox[3] = -1; myBox[4] = 0; myBox[5] = 0;
    myBox[6] = width; myBox[7] = length; myBox[8] = thickness;
    if (slate != null)
    {
        if (bodies.Contains(slate)) bodies.Remove(slate);

        slate = null;
        GC.Collect();
        GC.Collect();
        model.GraphicsRedraw2();
    }
    slate = modeler.CreateBodyFromBox(myBox);
    slate.Name = $"Cutting Slate {count++}";
    slate.Display3(part, 255 + 255 << 8, (int)swTempBodySelectOptions_e.swTempBodySelectOptionNone);
}
```

(4) 調整刀具位置

```
private void CutBodyMove(char kind, float distance)
{
    SetCutBodyMatrix(kind, distance);

    Body2 bd = slate.Copy();
    slate = null;
    GC.Collect();
    bool isOK = bd.ApplyTransform(matrix);
    Debug.Print(isOK.ToString());
    slate = bd;
    slate.Display3(part, 255 + 255 << 8, (int)swTempBodySelectOptions_e.swTempBodySelectOptionNone);
}
```

(5) 模型(b)與刀具(tool)進行 Boolean operation

```
object[] objects = b.Operations2((int)swBodyOperationType_e.SWBODYCUT, tool, out err);
```

(6) 計算切割後元件屬性

```
double[] oo = b.GetMassProperties(1.0);
```

(7) 產生分離特徵並儲存

```
swFeat = swFeatMgr.PostSplitBody((preSplitBodies), false, (originsToUse), (vBodyNames));
```

(三) API 函數

本研究使用的 API 函數如下

1. OpenDoc2

Opens an existing document and returns a pointer to the document object.

2. GetBodies2

Gets the bodies in this part.

3. GetMassProperties

Gets the mass properties of this body.

4. CreateBodyFromBox

Creates a temporary body from the specified box dimensions.

5. ApplyTransform

Applies a transform to this body.

6. SaveAs3

Saves the document to a different name.

7. Operations2

Performs add, cut, and intersect (unite, subtract, and interfere) operations between two temporary bodies.

8. PostSplitBody

Creates a split-body feature.

參、主要發現與結論

一、人機介面

人機介面使用方法如下

(一)使用者輸入模型文件

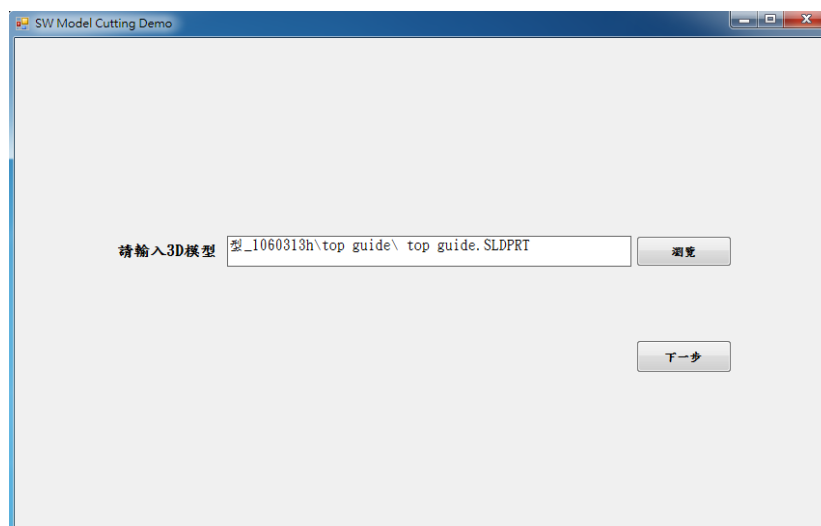


圖 28、輸入模型文件

(二)輸入切割相關參數



圖 29、輸入參數介面(矩形容器)



圖 30、輸入參數介面(圓筒容器)

(三)結果分析

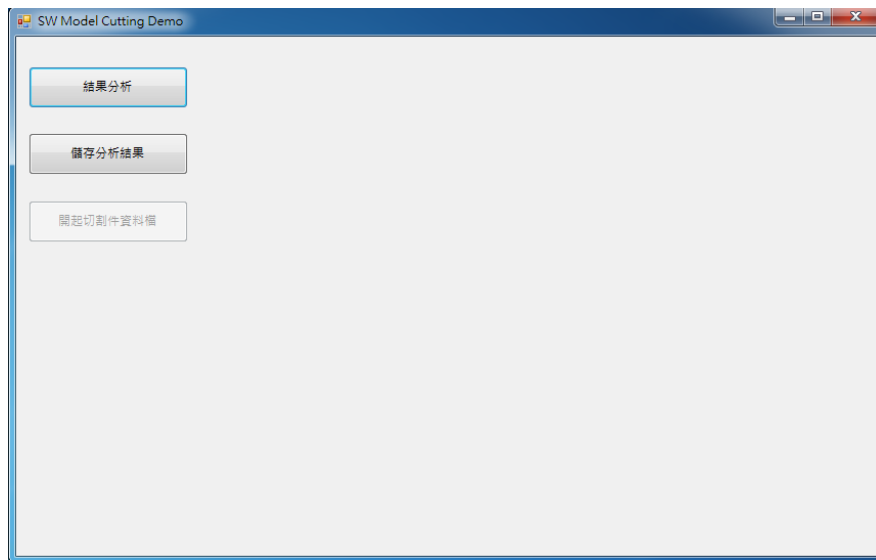


圖 31、結果介面

二、範例

在分析切割特性時，原先的模型也會被判斷，故切割特性數量會比切割件數量多 1。圖 32 為 SolidWorks 中 Top Guide 切割後呈現。圖 33 為 Visual studio 人機介面中 Top Guide 切

割後結果。圖 36 為 SolidWorks 中 Core Shroud 切割後呈現。

圖 37 為 Visual studio 人機介面中 Core Shroud 切割後結果。

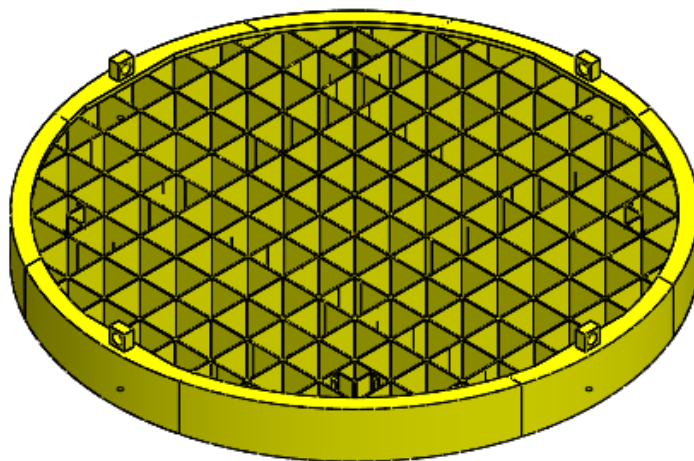


圖 32、SolidWorks 中 Top Guide 切割後呈現

SW Model Cutting Demo

結果分析

儲存分析結果

開啟切割件資料庫

切割結果計算

切割件數量: 9 件

切割總段面積: 296081.259 mm²

切割線總長度: 5815.924 mm

分析結果

編號	切割件檔名	切割件重量 (kg)	切割件體積 (m ³)
1	top guide1	505.9192	0.06323989
2	top guide2	442.2435	0.05528044
3	top guide3	415.2589	0.05190737
4	top guide4	482.8153	0.06035191
5	top guide5	362.4398	0.04530498
6	top guide6	552.2849	0.06903561
7	top guide7	306.9727	0.0383716
8	top guide8	500.1139	0.06251424
9	top guide9	352.9353	0.04411691
10	top guide10	3923.352	0.490419

圖 33、Visual studio 人機介面中 Top Guide 切割後結果

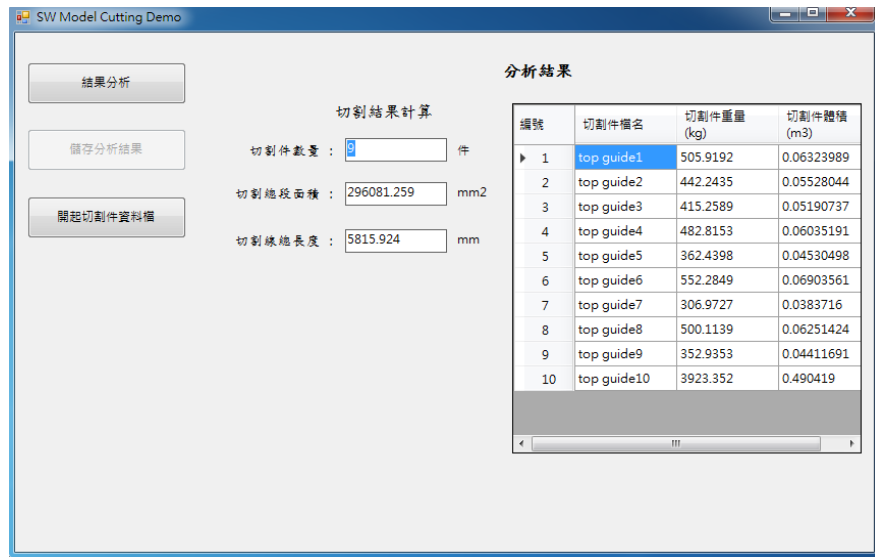


圖 34、儲存分析結果

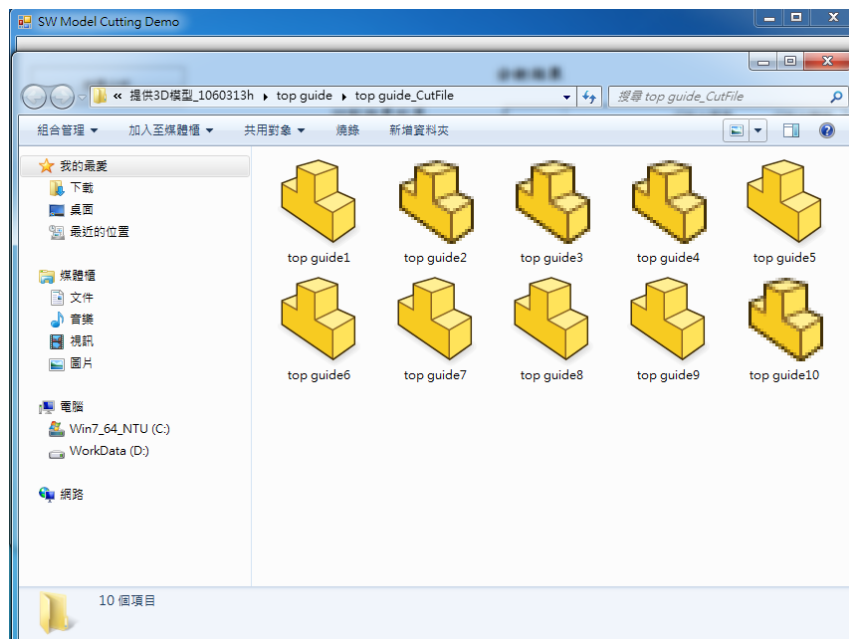


圖 35、開啟切割零件資料庫

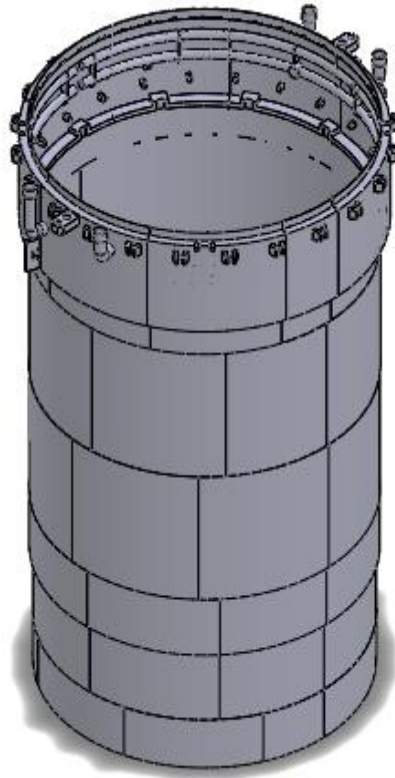


圖 36、SolidWorks 中 Core Shroud 切割後呈現

SW Model Cutting Demo

結果分析

儲存分析結果

開啟切割件資料庫

切割結果計算

切割件數量：41 件

切割總段面積：4470102.000 mm²

切割線總長度：111387.670 mm

分析結果

編號	切割件檔名	切割件重量 (kg)	切割件體積 (m ³)
1	ore structure assembly1	4027.52	4.02752
2	ore structure assembly2	105.3311	0.1053311
3	ore structure assembly3	105.3311	0.1053311
4	ore structure assembly4	105.3311	0.1053311
5	ore structure assembly5	105.3311	0.1053311
6	ore structure assembly6	96.36149	0.09636149
7	ore structure assembly7	100.8463	0.1008463
8	ore structure assembly8	82.90704	0.08290704
9	ore structure assembly9	105.3311	0.1053311
10	ore structure assembly10	105.3311	0.1053311
11	ore structure assembly11	105.3311	0.1053311

圖 37、Visual studio 人機介面中 Core Shroud 切割後結果

三、限制

在開啟模型的時候發現，有些模型並非單一零件，而是由多個個體組成。這會導致在 SolidWorks 中切割時，將獨立判斷這些個體，而在儲存切割後元件也會分別儲存，造成切割數量及儲存檔案上的錯誤。以 Reactor Pressure Vessel Shell 為例。圖 38 顯示 Reactor Pressure Vessel Shell 由 15 個個體組成。圖 39 為切割前與切割後之比較。由於並非單一個體，模型切割後儲存的檔案中，出現並非切割結果的檔案，造成切割數量及儲存檔案上的錯誤，如圖 40 所示。

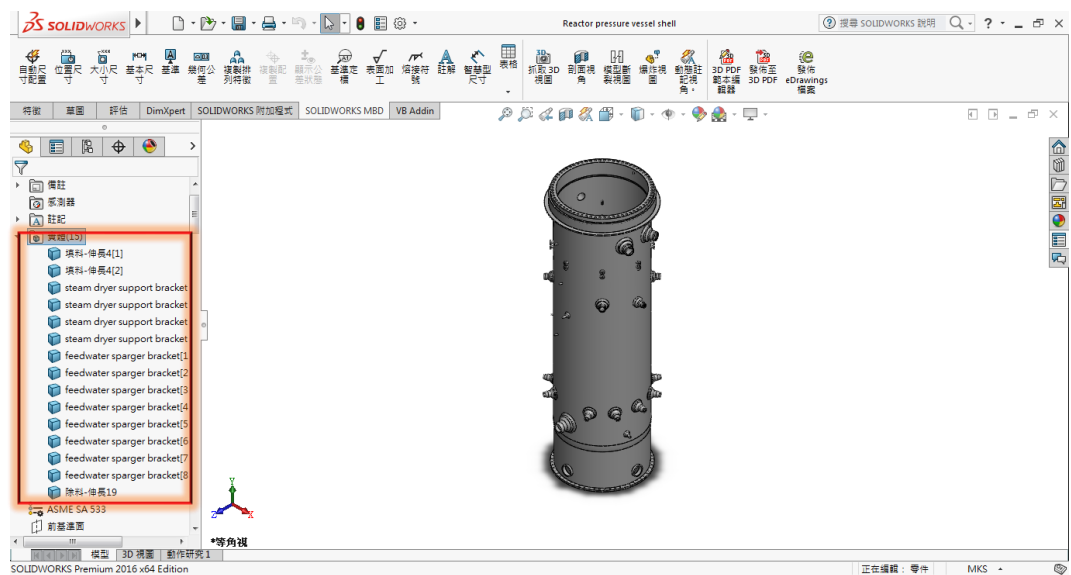


圖 38、Reactor Pressure Vessel Shell 由 15 個個體組成

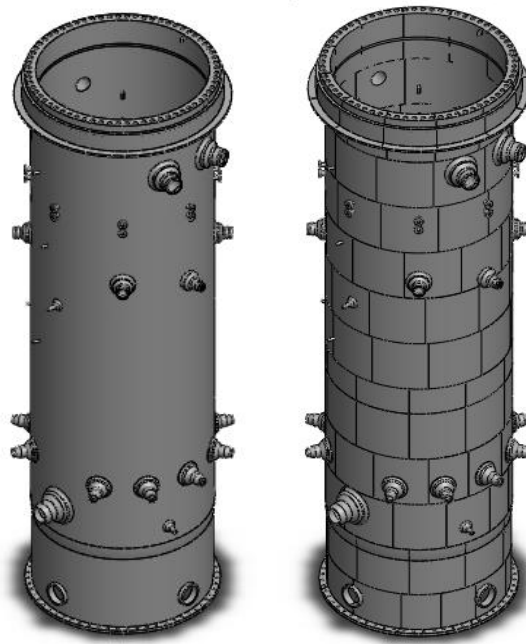


圖 39、Reactor Pressure Vessel Shell 切割前後模型

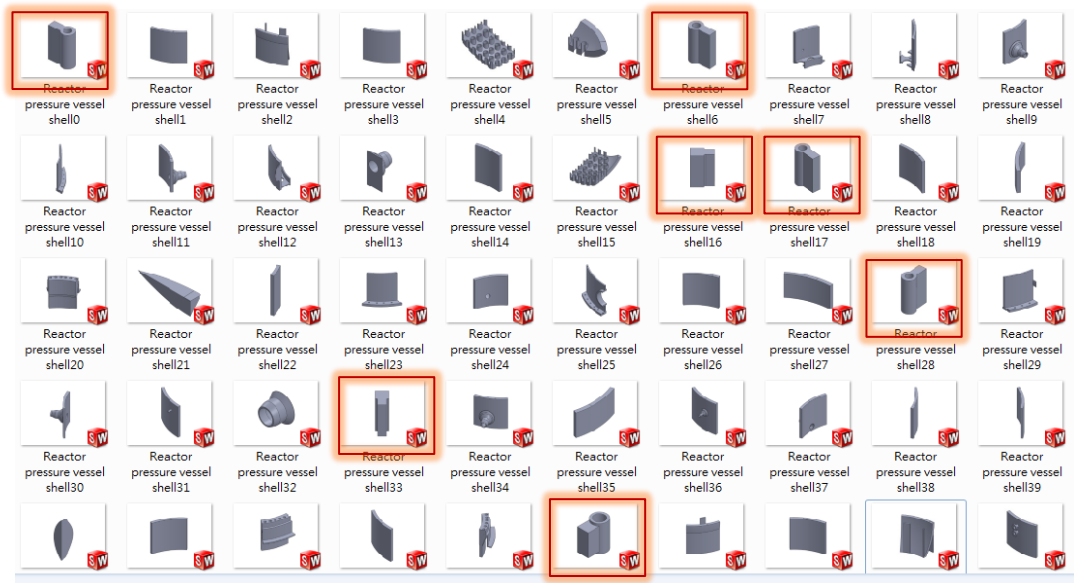


圖 40、Reactor Pressure Vessel Shell 切割後元件

目前的解決方法是利用“插入”——“特徵”--“組合”——“加入”使零件變成單一元件。但由於上熱屏蔽與 Reactor Pressure Vessel Shell 檔案中並非所有的零件都有交集，因此

這兩個檔案無法變成單一元件，如圖 41 所示。因此只有 Core Shroud 成功。最終的解決方法必須由繪製檔案的過程，確保模型為單一個體才能使切割過程不會出現錯誤。

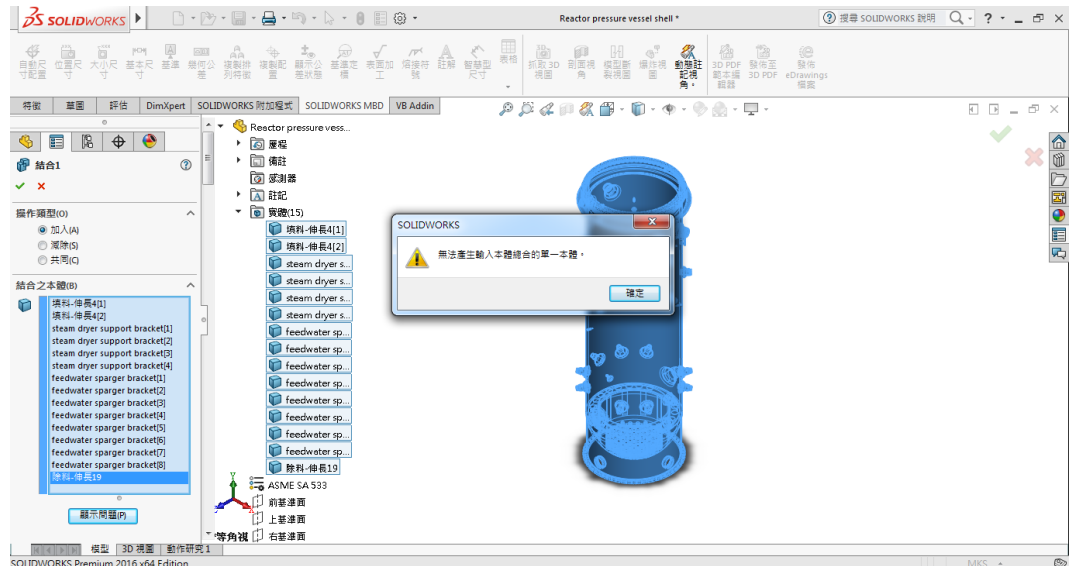


圖 41、模型無法被組合成為單一個體

四、結論

在這項研究中，我們藉由將核能所提供的 SolidWorks 模型由 sldprt 檔轉成 STL 檔，並且使用 C#讀取檔案資訊。將模型分成圓柱狀及圓餅狀，利用三角形頂點座標進行截面積的判斷。接著使用基因演算法，依照不同的裝箱容器需求，去求得切割截面積最小之切割結果。

圓柱狀模型採用 Y 軸切割以及角度切割；圓餅狀模型採用 X 軸切割及 Z 軸切割。將經由演算法得到的切割資訊透過 SolidWorks 的 API 函數，在 SolidWorks 中進行模型切割。切割

方法是利用模型與切割輔助平面進行 Boolean operation，並選擇模型作為保留主體。切割成果能呈現出切割數量以及切割面積，並分別計算每一個零件的重量，以及個別儲存零件。在測試過程中，基因演算法是快速且有效的，並且在編碼上，實數編碼比二位元編碼更快得到解。而截面積的計算對於演算法適應值的計算是有幫助的。但在模型並非為單一元件時會造成結果錯誤，在繪製模型時應確保此限制。

藉由此研究，對於核電廠的除役能夠減少刀具上的損耗，以便降低成本。未來可根據重量或裝箱輻射量進行切割最佳化。

肆、参考文献

- [1] B. Durak and D. T. Aksu, "A genetic algorithm based solution for the online cutting problem with defects," *2015 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, Singapore, 2015, pp. 1805-1809.
- [2] M. Spen, *Automating SOLIDWORKS 2015, Using Macros*, SDC Publications, 2015.
- [3] SOLIDWORKS API Help, <http://help.solidworks.com/2016/english/api/sldworksapiprogguide/welcome.htm>
- [4] Y. Chen, Y. Chen, S. Li and B. Li, "Research on precise 3D modeling of gears based on generating method," *2010 International Conference on Measuring Technology and Mechatronics Automation*, Changsha City, 2010, pp. 139-142.